An Argus Specialist Publication

MARCH 1985

85p

Your COMMODORE

YOUR BEST INDEPENDENT COMMODORE MAGAZINE

THE GAMES INVASION



WHAT'S IN A GAME—OUR SOFTWARE REVIEWS REVEAL ALL

WRITE YOUR OWN ADVENTURE GAMES: NEW SERIES

AN AWESOME INTERVIEW WITH JEFF MINTER

GIVING THE GAME AWAY—WIN THE COMPLETE CBS SOFTWARE RANGE

MUD, glorious MUD





ANIROG

Trade Enquiries: **ANIROG SOFTWARE 10 VICTORIA IND. PARK, VICTORIA ROAD, DARTFORD, KENT. (0322) 92513/8**
Mail Order: **8 HIGH STREET, HORLEY, SURREY 24 HOUR CREDIT SALES HORLEY 02934 6083**
Payment by: **P.O. - ACCESS - VISA - 50p POSTAGE & PACKAGING**

In the year 2525 A.D., time travel is finally perfected. Scientists of the project send you into the time void equipped with an armed, all terrain vehicle. Your objective is to assist mankind in his various battles : from the prehistoric age to the aftermath of the final conflict. Only a successful mission can assure your return. Features include 'Hi-Score table, two player option, two skill levels and continuous play facility.

Commodore 64 £7.95

Our COMMENT

This month in our special games issue the editor asks for fair play



PLAYING GAMES HAS HAD RATHER A bad press. "Stop playing games with me," we are told and recognise at once the accusation of deviousness and dishonesty. People who play games either meet deservedly sticky ends or they become the scheming heads of multi-national oil companies, like JR. And someone shot him.

Take the play, Sleuth, for example. Now I can't remember the plot exactly, but I can recall that there are two characters (or was it three?) who keep playing games with each other. First, one has murdered the other one's wife (or was it his mother?) who in turn (him, not his wife) dresses up as a policeman (or was he one all along?), then someone is shot (or was it stabbed?) but it's only pretend (I think!). All in all there's a lot of confusion and not just on stage.

So, playing games is either sinister or it's something which we are supposed to grow out of along with short trousers and spots. Sport is different, of course. It's O.K. for a grown man to spend an hour and a half on a Saturday afternoon knee deep in mud chasing after a funny shaped ball while trying to avoid fifteen even more fully grown men intent on separating his head from his body. Games are for those of us who haven't grown up!

Well, what about darts or snooker? Surely, they're games? Afraid not. The definition of a sport is an activity practised by men who are too big to argue with (darts and anything done by Geoff Capes) or if it's on the telly (snooker, darts and Geoff Capes). I will admit that dominoes and shove-ha'penny are two games played by grown men. However, they usually take place in the back rooms of pubs so filled with smoke that it's impossible to tell what's going on.

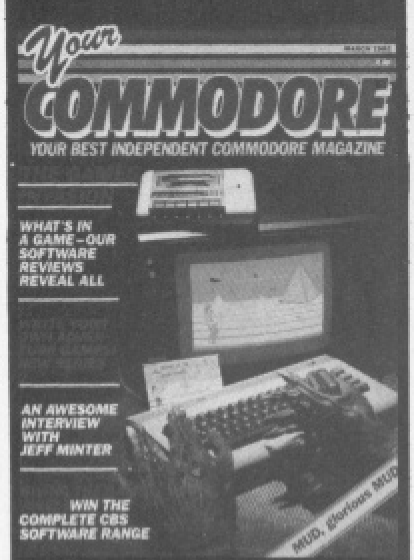
Unfortunately, some of these attitudes have rubbed off onto computer games. How often have you heard, "I'm a serious programmer. I don't play games." Or "I didn't buy you a computer just so you could play games." It seems that if you're 'serious' about computing, then you don't play games; if you bought your computer for some fun then you feel guilty about playing them and if you are under a certain age then you're not allowed to play them.

Well, I like playing computer games and I refuse to feel guilty about it. The good ones don't bypass the brain and even the bad ones are no more mindless than a weekly mud bath which leaves you with lungs like those of the fire-eater who sucked rather than blew. No one who has been following our Sense of Adventure series could possibly argue that computer games are not intellectually stimulating. The good adventure combines all the imaginative power of a novel with tests of reasoning and logic to stretch even the mathematical genius.

I recently researched an article on computers in schools and was surprised at the number of teachers who told me, "Of course, we don't allow anyone to play games." It may be preferable to get the class to run the 200 metres or do the high jump rather than waggle the joystick back and forth to make Daley Thompson do it (though I have my doubts). But there are a lot of games which require brainpower as well as manual dexterity. Take Impossible Mission (and if you win our competition this month you'll be able to do just that). Here is a game which is immediately attractive with brilliant graphics and speech but of such complexity that it will take a long time for you to crack it. Like all such games it teaches one of the most important truths about education, that practice is the path to improvement.

Programming is, of course, important and Your Commodore is doing its best to encourage it. Games are the best examples of good programming which are easily accessible to all of us. It seems to me, therefore, that if good programming is to be encouraged then the best way of doing it, is to show people exactly what can be achieved on the computer. Software houses should do this by producing the best possible games for the machine, ones which use all of its capabilities to the full. We are doing it by letting you know which games are the best and by printing some excellent examples ourselves. Cherry Picker in this issue is a long listing but for those of you with the patience and perseverence, this is a game worthy of any software house. You will also learn a lot about good programming just from typing it in.

Finally, games are entertainment. They are meant to be enjoyed. I hope that all you read in this issue will entertain you and perhaps even encourage you to start producing your own games to give pleasure to others. If you become good enough you'll be interested to read the article on getting your games marketed by a software company. Or you could send them to Your Commodore and share your talent with all our readers.



NUMBER 6
MARCH 1985
Editor: Wendy J Palmer
Deputy Editor: Kevin Cox
Editorial Assistant: Alison Hjul
Advertisement Manager: Mike Segrue
Advertisement Copy Control: Laura Champion
Chairman: Jim Connell
Origination: Ebony Typesetiing
Design: MM Design

Editorial & Advertisement Office
No 1 Golden Square,
London W1R 3AB
Telephone: 01-437 0626
Telex: 8811896

Your Commodore is a monthly magazine appearing on the first Friday of each month.

Distribution by: Argus Press Sales & Distribution Ltd, 12-18 Paul Street, London EC2A 4JS. Printed by: Alabaster Passmore & Sons Ltd, Tovil, Maidstone, Kent.

Subscription rates upon application to Your Commodore Subscriptions Department, Infonet Ltd, Times House, 179 The Marlowes, Hemel Hempstead, Herts. HP1 1BB.

FEATURES

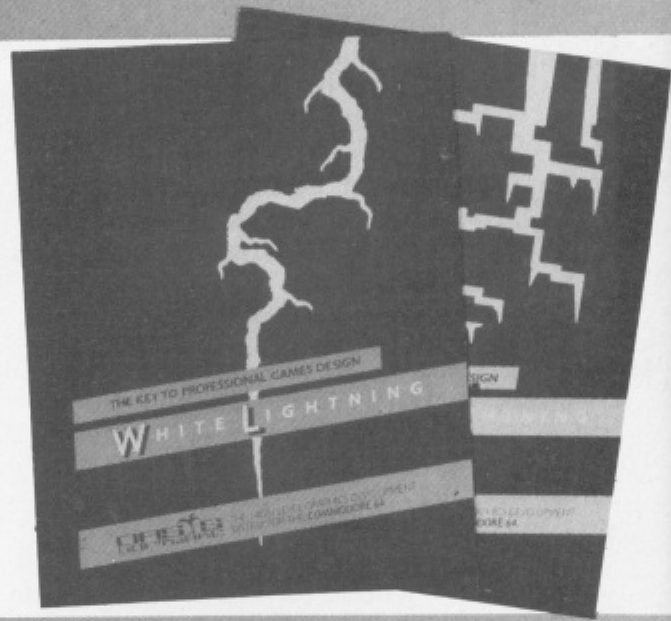


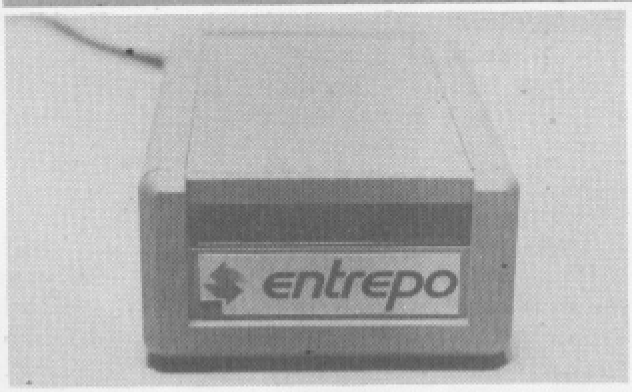
MINTERVIEW 6

What have sheep, llamas and chart-topping computer games got in common? The answer is, of course, Jeff Minter, the king of the hairies. He is the subject of this month's special interview.

LIGHTNING STRIKES THRICE 50

Oasis Software have recently released three software development packages for the Commodore 64. They are Basic Lightning, White Lightning and Machine Lightning. We hope you'll be struck by our reviews.





WAFERDRIVE 77

The Entrepo Waferdrive professes to be a tape storage device which is much faster than the standard tape? How true are these claims and what are its other capabilities? Read our review and find out.

STUCK IN THE MUD 86

MUD, standing for Multi-User Dungeon, is a multi-user adventure game on Compunet. To find out how it works, get stuck into our article.



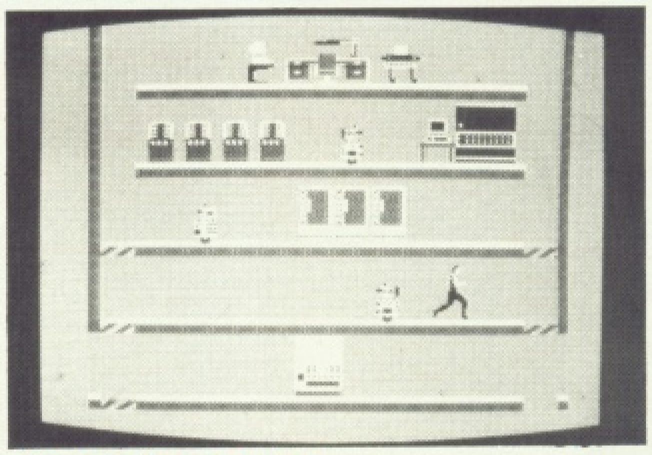
CONTENTS

SPECIAL FEATURES

SERIES

GAMES AND UTILITIES

COMPETITION



REGULARS

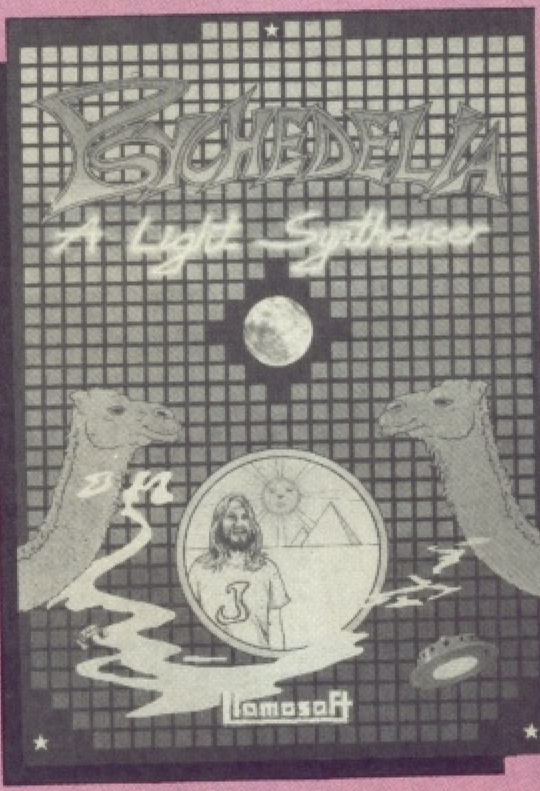
MINTERVIEW

Jeff 'The Yak' Minter is one of the best known personalities in games programming. Kevin Cox went to meet the hairy one.

EVERYTHING ABOUT JEFF MINTER IS hairy. His hair is hairy, his beard is hairy and his games are about hairy animals. His dog, an Afghan, was so hairy it quite startled me — I thought is was an undersized, overhaired llama — and I shan't mention the old adage about dogs starting to look like their owners. His kitten was not that hairy but obviously aspired to the condition. I can't remember now, but I'm sure that even the carpet was a thick shag-pile.

So, it was with some trepidation that I went to visit Jeff. Perhaps I should explain: I am not that hairy. It has been mentioned that I am thinning on top just a little, perhaps receding, certainly not balding. I am not touchy about it (*not much, Kojak – Ed*), but I had nothing to worry about. Fortunately, Jeff is not a man to be taken in by appearances unlike some I could mention (who is this Ed anyway?). He realised that I was as hairy as the next man, on the inside. And that's important, because hairiness for Jeff is a philosophy, it proclaims his individuality, his unwillingness to accept things at face value. It also means that he doesn't have to fork out a fortune for a haircut as often as the rest of us.

I visited Jeff at home in his family's bungalow in Tadley, a village half-way between Reading and Basingstoke. As soon as you arrive, you know Jeff lives there. On the wall is a large painted Llama. Inside they're everywhere. The room Jeff works in is a specially built extension packed with computer equipment: 64, Vic, C16 (he'd just bought one), Apple, Atari, MSX, BBC, QL plus his stereo compact disc and the video machines, including The Tempest and Atari's Star Wars. But the most striking feature of the room is the mural all along one wall. And the subject matter? Llamas, of course. And then the Llamas on top of the monitors —



fluffy ones, plastic ones, metal ones. Not to mention the camels and the alpacas.

It seems that Jeff can't remember when this obsession with large ruminant quadrupeds began. While still at school one of his first games, programmed on an 8K Pet, was called Vicuna.

In those days, he used to get up at 6 o'clock in order to get to school and start programming. There was only one machine and time on it was limited. It took him two or three months to learn BASIC, but he soon tired of its limitations, so he taught himself machine code.

"It took me three days to learn Z80 machine code."

Getting up at six takes its toll, though, and he saved for 6 months to buy a ZX80. By this time, his talent was obvious and he told me, "It took me three days to learn Z80 machine code." Gulp.

Unfortunately, no-one recognised his gift at university, where only a third of his course was computing, so he left after a year. Then, after a couple of spells working for dk'Tronics (he developed a Graphics ROM for the ZX81) and for Interceptor Micros (where he produced versions of Amidar and Defender), he started on his own. Gridrunner arrived and the real Llamasoft was born.

No more getting up at six now. "I work whenever I feel like it," he told me. But his work-rate is prodigious. Just think of the number of games, all different and innovative, which he has produced in the last 18 months: Hover Bovver, Hellgate, Revenge of the Mutant Camels, Sheep in Space ('my personal favourite scrolling shoot-em-up' he said) and Ancipital. And that's just a selection. Plus, he is now producing a regular magazine, the Nature of the Beast, all done on the wonderful Macintosh. It's very readable, very controversial in its opinion of games (and magazines), and a lot of fun. If you haven't seen a copy, write to Jeff at Llamasoft.

When I met Jeff he hadn't exactly been slacking. "I've never worked so hard in my life," he said. He had just spent two weeks on a brand new program, really a brand new idea, Psychedelia. He had been working on a game when the idea came to him, and once that had happened he dropped everything to complete it. In two weeks it was finished, not just on the 64, but on the Vic and C16 as well.

So what is Psychedelia? It is not a megagame. There are no ladders, no ramps, no bullets, no score, no lives, no aliens, no smooth scroll, no sprites, no lasers. Not a lot of anything, in fact, I thought. Just shows how wrong you can be. Jeff turned down the lights, put on Thomas Dolby (the one with the Llama on the album cover!), picked up the joystick and started. The plain white pixel in the middle of the screen burst into life. Colour was everywhere, in shapes, patterns, movement. Pschedelia had me hooked, I couldn't take my eyes off the screen. I'll wear a flower in my hair. *(What hair? Ed)*

Psychedelia is a light synthesiser. It is designed to be 'played' with a joystick, in much the same way as you would play a music synthesiser. The keyboard offers a variety of options such as pre-defined shapes (including a Llama), symmetry, colour choice, shape sampling etc. The joystick lets you create to the music of your choice, to interpret in light the sounds you most enjoy. Anyone can do it, and skilfully too. If I have any criticism, it is that the results that a novice can get are so effective that I cannot see how practice will necessarily improve them. You cannot sit down at a music keyboard and

just play a tune. But perhaps I haven't seen what a really skilful player can do. After all, when I saw it, Jeff was the world's most experienced user, and he'd only been doing it for two weeks!

The program's possibilities are endless. Think of creating your own audio-visual extravaganza for a party — the 64 version will save up to an hour to tape. Or you can just sit in a dark room to enjoy the experience.



"I work whenever I feel like it."

Now that Psychedelia is out of his system, Jeff can return to the game he abandoned. It is called Mamma Llama, and the stars are three llamas, a mother and two youngsters. It is much more in the tradition of Minter games — smooth scroll, excellent animation, bullets, aliens, and of course llamas. It is clear, however, that Jeff feels the time for such games is passing. A lot of the innovations he began, everybody has now copied. Smooth scrolling, for example. He was the first on the 64. He had seen it on the Atari and then he "sussed it out from the Commodore manual." He feels that Mamma Llama is his last game to use the technique. It has been done to death and he is very scathing of programmers like Tony Crowther whom he feels use it for no other reason that it's there. It goes against his first principle of games writing: "Originality is where it's at," he said.

So whose games does he like? Taskset was a name which came in for a lot of praise. And he also showed me two imported games for the Atari, developed by Lucasfilms, called Behind Jaggi Lines and Ballblaster. They're good, very good. Fingers crossed that they come over here for the Commodore.

Mamma Llama may not be as great a departure as Psychedelia, but it has all the quirkiness that sets a Minter game apart from the rest. As Jeff said, "It takes a certain type of mind to develop games — freaky". He certainly has that. Our family of Llamas travel through Peru, to Egypt (got to get the camels in somewhere) and to the moon. On the way, look out for references to Jeff's favourite radio station KMEL106 FM from Los Angeles (symbol: a camel) and to his favourite drink, Inca Cola, a yellow, Peruvian version of the more famous original.

He discovered Inca Cola on his trip to Peru last year. (For a full report, read The Nature of the Beast 3). While I was with him, he showed me his photos. And if I can't yet share his fascination for Llamas (every pic had at least one in it) I can certainly see why he would want to visit their homeland. He travelled all round the country, to Lima the capital, on a railway at 14,000 feet above sea level, to the mountains surrounding the great Inca ruin of Machu Pichu. Jeff is not a programmer who is manacled to the computer. Like a good writer or musician he is open to all sorts of influences and they are reflected in his work.



His energy is limitless; he goes running every day and he also skis. If there's one word which sums up his attitude, it's enthusiasm. A lot of people enjoy his games because they realise he is a programmer who likes playing games himself — and his own games are the ones he most likes playing. He likes to hear from people about what they think of his games (and other people's) and, above all, he likes going to shows to meet the people who share his interest. I saw him at the last PCW show, not selling like everyone else, but completely engrossed in a two-handed game of Ancipital with a fellow enthusiast.

"It takes a certain hype of mind to develop games — freaky."

He is not commercially minded and doesn't like the new atmosphere. He is prepared to put free programs up on Compunet, for example, and nearly did so with Psychedelia until persuaded by his mother that it might not be a wise decision. He liked the early days when everyone was an enthusiast and understands the tragedy of good programmers being hyped out of the market nowadays. But I cannot share his belief that the old days will return, "Maybe this summer will sort out the sheep from the goats," he said somewhat ironically.

Jeff Minter is the best known programmer in this country. A superstar? I asked him. "I don't really think of myself as a superstar," he said. "My idea of being a superstar is to play a light synthesiser at a concert." I hope it happens. He is a very modest superstar, the best kind to be.

We've got the TALENT



and our games prove it!

Amazing graphics, fast and furious action, challenging strategy, compelling adventure – this first wave of games from TALENT has got the lot!

Written by professional computer scientists using powerful new programming techniques (which leave machine code standing), these games have pushed home micros to the very limit.

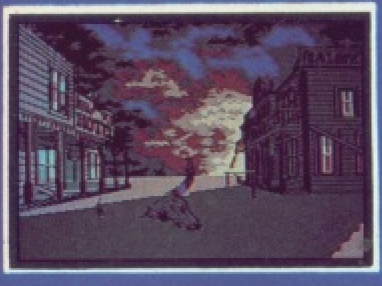












KALAH

One of the oldest games in the world! Originally played by Bedouins using pebbles and holes scooped in the desert sand, this strategy game has absorbed the mind of man for thousands of years. Superb graphics and music, with a choice of levels and depth of search. Easy to learn, fiendishly difficult to beat!

Commodore 64 Cassette £7.95 Disk £9.95

LASER REFLEX

A new angle in alien zapping! Against a brilliant backdrop of stars, you must defend your asteroid base from wave after wave of swooping mutating starships. Your defence – a powerful laser deflected against a giant mirror. How long can you hold out before your defences crumble?

BBC Model B Cassette £7.95
Electron Cassette £7.95

ARCHIPELAGO

Dare you venture into the ancient undersea labyrinths which link the islands of the Archipelago? Fabulous jewels await you there – but the guardians of the treasure mark your every move, waiting to deal a swift and terrible vengeance. How long can you survive?

A fast-moving, action-packed game with brilliant graphics, animation and music.

Commodore 64 Cassette £7.95 Disk £9.95
(joystick needed)

WEST

A compulsive adventure game set in a ghost town in the Wild West. Your task – to flush out a gang of bank robbers and retrieve their loot – with only your trusty steed and your six-shooter to aid you!

Test the speed of your draw, your skill at cards, your powers of survival!

The first of a stunning series of adventure games from Allan Black, played in real-time.

Commodore 64 Cassette £9.95 Disk £12.95
BBC Model (text only) Cassette £7.95
Electron (text only) Cassette £7.95



SOFTWARE FROM SCOTLAND

Talent games should shortly be available from your local software retailer or post off the coupon now.

PANORAMA (H)

An outstanding graphics package for the Commodore 64. Features include: free-hand sketches, circles, ellipses, block move, copy and reverse, magnification choice of cursor, joystick or key control. PLUS texture definition and texture and colour fill. PANORAMA (H) uses the full colour facilities of the Commdore 64.

Commodore 64 Cassette £17.95 Disk £19.95

The programme is menu-driven and comes with a comprehensive illustrative instruction booklet.

TALENT

Curran Building, 101 St James Road
Glasgow G4 0NS Tel 041 552 2128

INSTANT CREDITCARD 24 HOUR ORDERING SERVICE TEL 041 552 2128

TO: TALENT COMPUTER SYSTEMS FREEPOST (no stamp required) GLASGOW G4 0BR
Please send me the following items

COMMODORE 64	Quantity		
PANORAMA (H)	—	Cassette	(£17.95)
PANORAMA (H)	—	Disk	(£19.95)
ARCHIPELAGO	—	Cassette	(£ 7.95)
ARCHIPELAGO	—	Disk	(£ 9.95)
KALAH	—	Cassette	(£ 7.95)
KALAH	—	Disk	(£ 9.95)
WEST	—	Cassette	(£ 9.95)
WEST	—	Disk	(£12.95)

BBC MODEL B	Quantity		
LASER REFLEX	—	Cassette	(£ 7.95)
WEST (text only)	—	Cassette	(£ 7.95)
ELECTRON			
LASER REFLEX	—	Cassette	(£ 7.95)
WEST (text only)	—	Cassette	(£ 7.95)

Total (includes postage and packing)

I enclose a PO/Cheque to the value of £ ____________

Please debit my ACCESS ____________

Name (block capitals) ____________

Address ____________

____________ Post Code ____________

Signature ____________ Tel ____________

Please allow up to 28 days for delivery. We shall not bank your remittance until your order has been despatched.
Dealer enquiries welcome. Contact Veronica Colin on 041-552 2128.

Data Statements

"We do not react to Atari" says Commodore boss

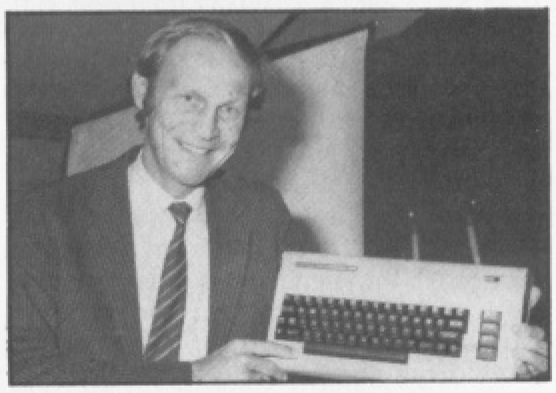
JACK TRAMIEL, FORMER BOSS OF Commodore, and now the head of Atari breezed into the country recently to announce his new company's range for this year. He announced his intention to 'build a new line, a more advanced line for the masses.'

His scheme is based around four different families of machine. The first are the games machines to replace the existing range. The second are the 8-bit machines, 4 of them: 64K, 128K, a machine with advanced music capabilities, and a portable with built-in 5 inch screen. Thirdly, there is a 16-bit machine based on the 68000 with hi-graphic capabilities. Finally, there is a 32-bit complete workstation for around £1000, to be launched next July.

Commodore's reaction to the announcements was understandably muted. Howard Stanworth, Commodore UK's General Manager told me, "Our view is that announcements are announcements, products are products." He would not be drawn on Commodore's new product line-up. "We do not react to Atari — our decisions will be based on our own product strength." Commodore's next announcement is scheduled for January when we should learn more about the PC-compatible and the Z-machine.

Jack Tramiel





Howard Stanworth

Finally, how does Commodoree view Atari's policy of selling the 800-XL at £130? Howard Stanworth again, "The customer has demonstrated that the consumer is not interested in distress marketing." Still, it all promises to be an interesting battle. Commodore may not be rattled but they must be aware that Tramiel is not finished yet. Otherwise why would 10% of his head office staff all have come from Commodore Business Machines.

Death of the datasette?

PACT INTERNATIONAL LIMITED HAVE released a cassette interface which allows Commodore 64 and VIC 20 programs to be saved or loaded from an ordinary domestic cassette recorder. It is called the PANDA 20/64 and retails at £17.99. This eliminates the need to fork out £45 for a dedicated Commodore recorder when you might already own a cassette recorder.

The interface features a special phase switch that enables it to cope with different types of recorder and tape quality. It also has a couple of small lamps (LED's) which indicate when a program is being saved to or loaded from cassette. It even loads 'turbo' tapes.

The product is available from most electrical, television, hi-fi, video and computer shops throughout the UK which stock PANDA branded products.

Pact International Limited, P.O. Box 50, Peterborough, England; tel. 0733-233600.



Fangs ain't what they used to be

IT IS TRANSYLVANNIA. THE YEAR IS 1880. In the village inn the talk is of a beautiful young maiden who has disappeared. The prime suspect is the secretive Count who lives in the nearby castle. The villagers plead with you to go to the castle to rescue the girl, though they know that no man has ever returned alive before.

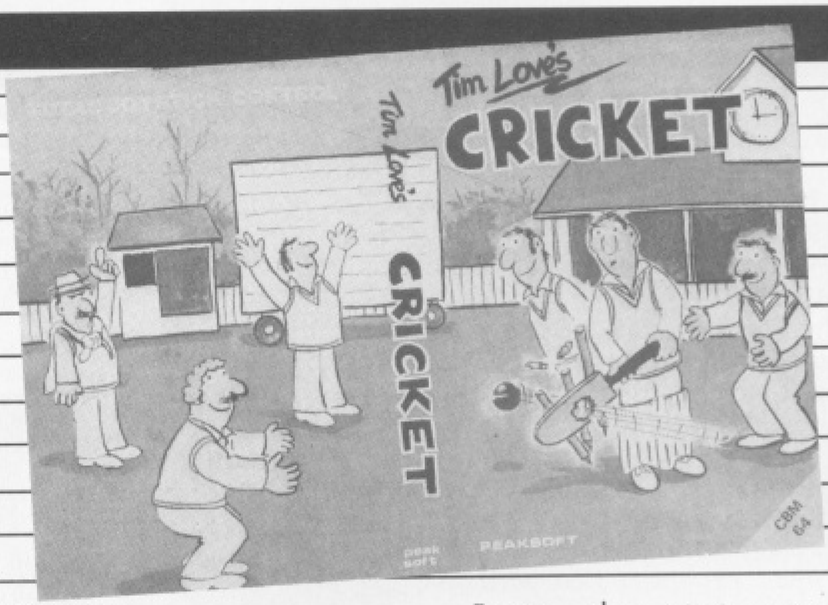
This is the outline story of Castle of Terror, the new graphic adventure game from Melbourne House, What I want to know is, why does everyone suspect the poor Count? Perhaps he's just a little eccentric. What's wrong with enjoying the occasional Bloody Mary? Not everyone likes garlic and perhaps it is more comfortable to sleep in a coffin. Have you tried it?

However, if you think there's more to this than a simple course of dental treatment can cure, then for £9.95 this could be the game for you. It promises to be an adventure to get your teeth into.

Melbourne House, Castle Yard House, Castle Yard, Richmond, Surrey.

Data Statements

Howzat!



WILLIS BE ANOTHER NAFF CRICKET simulation, I thought. I'll BOYCOTT it. I'll MARSH right over to the programmer and tell him. But when I SOBERS up, I realise what a COWDREY I am. I'll give it a GOWER.

The simulation in question is Tim Love's Cricket. It comes from Peaksoft, price £8.95 for the CBM 64. The copy we received is pre-production and has few anomolies which will be ironed out later. I quote from the blurb: "In production copies, the fall-back team will be 'England' and not 'England*, Gatting will not have a distressing tendency to take over the wicket-keeping in the fielding sequences (can't do any harm — Ed.), and it will be impossible to dismiss any Nottinghamshire batsman for less than 100." And I thought it was supposed to be a real-life simulation.

I haven't yet had a chance to give it a test, but I'm sure it won't have reached rock BOTHAM.

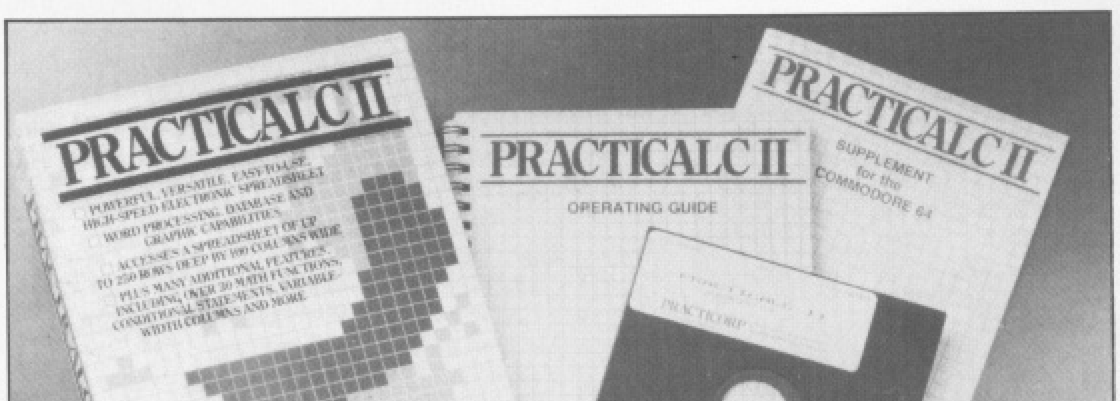
From the statement about Nottinghamshire, you can probably guess where Peaksoft is based: 48 Queen Street, Balderton, Newark, Notts.

Practicalc II

PRACTICORP LTD HAS JUST RELEASED A new, low-cost software package for the 64 which incorporates spreadsheet, word processing and database functions. All can be incorporated into a single working document.

It is Practicalc II and costs £69.95. Looking at the spec, it seems to offer the small business user a fair range of features: the database, for example, has automatic alphabetic and numeric search and sort and the word processing has the capabilities for writing, editing and justifying text.

Practicorp Ltd, Goddard Road, Whitehouse Industrial Estate, Ipswich, Suffolk.



A good cause



IF EVER THERE WAS A TIME TO BUY Commodore equipment, then now is it. Commodore will donate 25p to the Save the Children Fund for every completed guarantee card returned to them by February 28th. The target is £10,000 which will go to Ethiopia as famine relief.

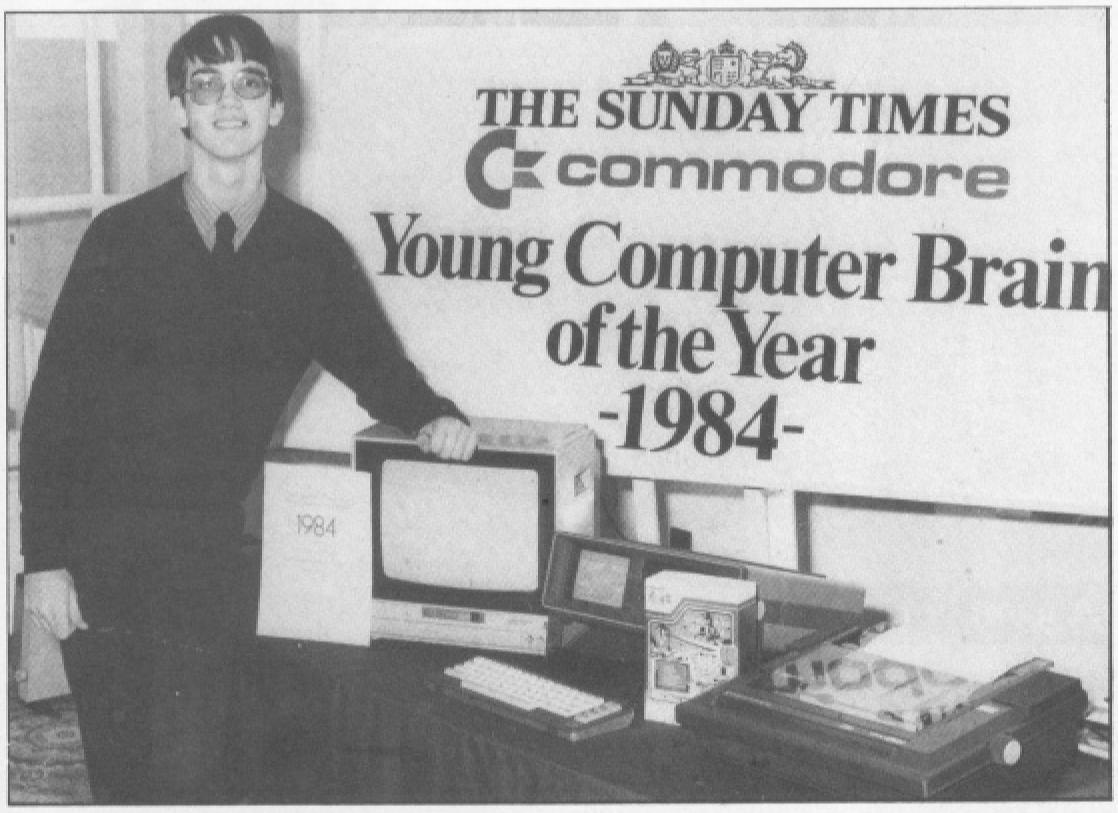
If you buy a Commodore, remember to return the card. Time is pressing.

Data Statements

All at sea

A CAMBRIDGE SCHOOLBOY, PETER Chase, is the Young Computer Brain of 1984. The competition is organised by Commodore in conjunction with the Sunday Times. Entrants are invited to come up with new and imaginative ideas for using computers in ways which could benefit society.

Peter, who is only 15, invented a system called Coastel. It is a computer/modem system to aid sailors and coastguards, with the aim of making sailing safer, navigation easier, rescue quicker and more detailed information available to sailors. His prize was £2,000 worth of computer equipment, including an SX-64.



Summer and winter

THE NEXT OLYMPIC GAMES MAY NOT be until 1988 but you won't have to wait that long for the next installment of Summer Games from American software house, Epyx. The next Summer Games will feature fencing, equestrian events and possibly another diving competition. However, the exact choice of events has not yet been decided.

In September this will be followed by Winter Games which should feature bobsleigh, ice skating, grand slalom and even barrel jumping whatever that is.

The last Summer Games was released under license to Quicksilva but the new ones will appear under the CBS Software label. For more news on CBS look out for the great competition for CBS Software in this issue.

The C16 takes off

CRAIG COMMUNICATIONS HAVE JUST launched a flight simulation program for the C16. It is called Flight Zero One Five and is based on the one of the same name for the VIC. There are five skill levels and all the usual features: instrument display, artificial horizon, status reports, pilot rating, reverse thrust on landing (sounds nasty), and realistic sound effects — "This is your captain speaking. Drinks will be served in 5 minutes." Tickets cost £5.95.

Craig Communications, PO Box 46, Basingstoke, Hants.

Black Thunder

QUICKSILVA LTD, AND TONY Crowther, Director of Wizard Development Company, have formed an agreement for Quicksilva to market two of Crowther's games. The first, for the Commodore 64, is called Black Thunder.

As the hero, Super Human Crow-Ther, you must fight the evil Wizard as you travel the roads of a surreal landscape in your futuristic car. The action in the top half of the screen scrolls smoothly as radar shows you your progress. The opposition's progress is shown in the bottom half of the screen. Black Thunder features software produced speech and will also drive the Currah Speech Unit.

Black Thunder is available on disc for £12.95 and cassette for £7.95.

Quicksilva, Palmerston Park House, 13 Palmerston Road, Southampton, Hants; tel. 0703-26515.



Data Statements

Pencil it in

THE DESIGNER'S PENCIL FROM ACTIVISION enables you to draw graphics on the screen and create sounds and tunes by using the joystick. Activision claim that this innovative product, designed by Garry Kitchen, virtually eliminates 'computer control and complex programming design' while allowing the user to create pictures and sound 'with the simplicity and fun of playing a game'.

It is supposedly impossible to enter a command which the computer doesn't understand since a Command menu contains every command likely to be used. To design a game, the user need only position a cursor arrow next to the command of his or her choice and press the 'Fire' button.

The Designer's Pencil is also educational as it teaches people the workings and potential of their home computer.

The Designer's Pencil retails for £11.99 (cassette) and £19.99 (disc) on the Commodore 64.

For further information tel. 0628-75171.

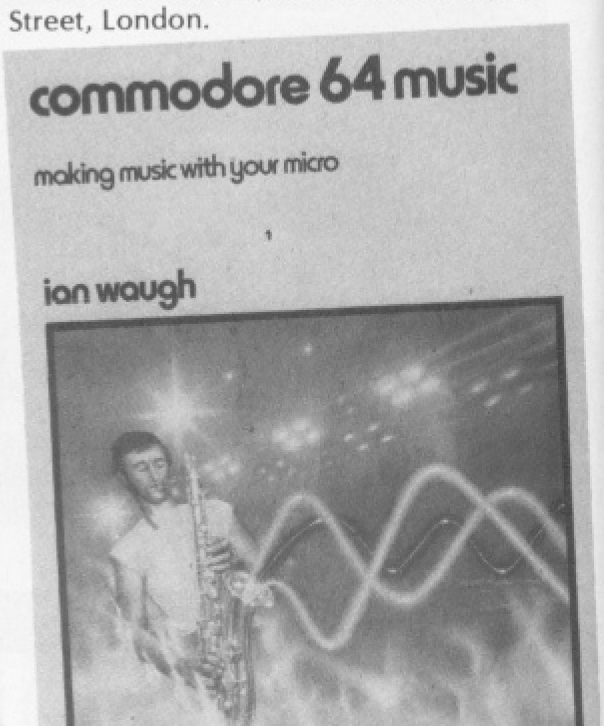


Play it again, Sid.

TWO NEW BOOKS ABOUT THE musical capabilities of the 64 have just been published by Sunshine Books. The first, Electronic Music on the Commodore 64 by Mark Jenkins (£6.95), explains the SID sound chip and includes music routines which can be included in your own programs, in whatever style of music you want.

The second, Commodore 64 Music is written by Ian Waugh, a professional musician who has already written a book on music on the BBC. All the programs are written in BASIC and they allow you to produce chorus, echo, polyphonic music, microtonal scales and even sound effects like seagulls and foghorns. Vaughan Williams' Sea Symphony here I come.

Sunshine Books, 12/13 Little Newport Street, London.



Into battle

THEY'RE A BELLICOSE LOT AT U.S. GOLD Hot on the trail of Raid Over Moscow comes yet another 'war adventure' — 'Blue Max'. The 'Blue Max' of the title is Max Chatsworth. The action takes place in the cockpit of his plane during the First World War as he battles against the Axis powers. U.S. Gold's hero must shoot down enemy planes, bomb targets and strafe gun emplacements and tanks. To complete the mission, Max must make a final assault on three specially marked targets within the enemy's city.

The player has to master a series of flying skills and bomb targets at the same time as keeping track of fuel, altitude and speed. The enemy retaliates with anti-aircraft fire — and, beware any damage and fuel leakage from Blue Max's plane.

U.S. Gold believe that the 3-D diagonally scrolling screen gives constant realistic action.

Blue Max is available on cassette or disc. It retails on the Commodore 64 for £9.95.

A game for the new year

BIG BROTHER'S STOPPED WATCHING us and George Orwell is just another writer. It's 1985 — or, at least it is in the latest offering from Mastertronic. The aim of the game is to guide a small spacecraft through a series of caverns to collect stores of nuclear plasma. Having done this, the final task is to locate and collect the fusion core from the last and most difficult cavern. As with all Mastertronic games, 1985 costs £1.99.

Mastertronic, Park Lorne, 111 Park Road, London NW8 7JL; tel. 01-402-3316.

Alien Hotline

ARGUS PRESS SOFTWARE HAVE BEEN inundated with calls from frustrated Alien fans. Since there are so many people out there who obviously can't solve the game without further clues, an Alien hotline has been set up.

Callers' problems range from being unable to get into the shuttle to the best way to kill the alien. And they're offered such handy advice as to 'watch the cat' or 'get the cat box'.

If you are really in the depths of despair phone Argus Press Software on 01-437-0626.

TOP 20 Software
Compiled by Gallup

COMMODORE 64

TITLE	PUBLISHER
1. Ghostbusters	Activision
2. Daley Thompson's Decathlon	Ocean
3. Raid over Moscow	US Gold
4. The Staff of Karnath	Ultimate
5. International Football	Commodore
6. Zaxxon	US Gold
7. Beach Head	US Gold
8. Booty	Firebird
9. Combat Lynx	Durell
10. Summer Games	Quicksilva
11. Hunchback	Ocean
12. Jet Set Willy	Software Projects
13. Bruce Lee	Gremlin Graphics
14. Blue Max	US Gold
15. Fighter Pilot	Digital Integration
16. Football Manager	Addictive Games
17. Pyjamarama	Mikro-Gen
18. Monty Mole-Wanted	Gremlin Graphics
19. Chiller	Mastertronic
20. Tapper	US Gold

Retail sales for the month ended Dec 31 1984



VIC 20 Top Ten

TITLE	PUBLISHER
1. Perils of Willy	Software Projects
2. Hunchback	Ocean
3. Micky the Bricky	Firebird
4. Jetpac	Ultimate
5. Psycho Shopper	Masteronic
6. Vegas Jackpot	Mastertronic
7. Flight Path	Anirog
8. Bullet	Mastertronic
9. Snooker	Visions
10. Duck Shoot	Mastertronic

Retail sales for the month ended Dec 31 1984

Compiled by Gallup for the industry's weekly trade magazine, Computer and Software Retailing. For details contact John Sorrenti, Computer and Software Retailing, No. 1 Golden Square, London W1R 3AB. 01-437 0626.

Use your memory and judgement to guide the tank driver through the minefield and back safely to his tank in Paul Randall's game for the unexpanded VIC 20.

MINE FIELD



The mines appear at the start and then disappear. You must remember where they are and do your best to avoid them: You must go around the trees which are scattered around the screen. You control the main either by using a joystick or the keyboard. When you reach the tank, the game starts again with more mines but less time to study them.

To play the game, type in the first, smaller program and, after checking the data, save it. Then type in the other program and save it (don't RUN it as the program contains machine code so it might crash). If it crashes, check the data in program 1.

Program Listing 1

```
1000 PRINT"■■       ■MINEFIELD■"
1001 PRINT"■■     BY P.RANDALL"
1002 PRINT"■    VIC 20 3.5 K"
1003 FORJ=1TO15
1004 POKE36878,J:POKE36875,128+J*5
1005 FORR=1TO100:NEXTR
1006 NEXTJ:POKE36875,0
1010 PRINT"■ YOU HAVE BEEN LEFT"
1012 PRINT"■  IN A MINE FIELD"
1014 PRINT"■ YOU HAVE TO GET BACK"
1016 PRINT"■ TO YOUR TANK WITHOUT"
1018 PRINT"■ WALKING OVER A MINE"
1020 PRINT"■  YOU ARE SHOWN THE"
1022 PRINT"■MINES AT THE START BUT"
1024 PRINT" THEY THEN DISAPPEAR"
1026 PRINT"■       ■HIT A KEY■";
1028 GETL$:IFL$=""THEN1028
1030 PRINT"■■       ■MINEFIELD■"
1032 PRINT"■■ MOVE YOUR MAN EITHER"
1034 PRINT" ■USING A JOYSTICK OR:"
1036 PRINT"■  ■A■-UP     ■Z■-DOWN"
1038 PRINT"■  ■V■-LEFT   ■B■-RIGHT"
1040 PRINT"■■H■-LETS YOU LOOK AT THE"
1042 PRINT" MINE FIELD AGAIN BUT"
1044 PRINT"■  EXPENDS 100 POINTS"
1046 PRINT"■YOU CAN'T WALK THROUGH"
1048 PRINT"THE TREES OR BOUNDARY"
1050 PRINT"■       ■HIT A KEY■";
1052 GETL$:IFL$=""THEN1052
1060 PRINT"■■■■■■■■■■"TAB(7)"■LOADING■"
1070 J=7416
1075 READA
1080 IFA=300THEN1095
1085 POKEJ,A
1090 J=J+1:GOTO1075
1095 GOTO1500
1096 DATA0,195,195,255,255,195,195,0
1097 DATA0,0,0,0,0,0,0,0
```

Program Listing 1 (cont.)

```
1100 DATA63,109,219,183,237,219,162,252,0,60,102,90,90,102,60
1105 DATA0,16,56,16,56,84,16,40,68,0
1110 DATA0,0,19,11,5,3,3,0,0,0,0,128,3,255,255,7,31,63,127,255,127,63
1115 DATA31,131,192,240,252,255,254,248,192,0
1120 DATA0,0,0,0,0,192,224,0,0,0,0,0,255,15,13,192,192,192,192,192
1125 DATA240,252,255,3,3,3,3,3,15,63,255,255
1130 DATA255,254,252,220,196,192,192,255,255,127,59,35,3,3,3,224
1135 DATA248,254,254,254,254,254,255,7,31,127,127,127,127,127,255,60,300
1500 DATA120,169,0,141,20,3,169,***,141,21,3,88,96,169,0,141
1510 DATA19,145,141,34,145,173,17,145,41,31,74,74,133,144,173,17
1520 DATA145,41,32,74,5,144,133,144,173,32,145,41,128,74,74,74
1525 DATA74,5,144,73,31,133,144,169,255,141,34,145,76,191,234,234
1530 T=0:S=673
1540 FORI=0TO63
1545 READA$:IFA$="***"THENPOKES+I,(S+13)/256:POKES+I-5,(S+13)AND255:NEXT
1550 POKEI+S,VAL(A$):T=T+VAL(A$):NEXT
1565 POKE198,0:PRINT"[illegible]"CHR$(34)""CHR$(34):POKE198,2:POKE631,19:POKE632,131:N
EW
READY.
```

Program Listing 2



```
1 POKE52,28:POKE56,28
5 POKE36869,255
9 PRINT"[illegible]ENTER SKILL  (1-5)";:INPUTSK:SKILL=SK*1000:PRINT"[illegible]"
10 POKE36879,136:C=7703:M=10:N=3
15 P=8152:Q=38872:L=1
16 SYS673
17 GOSUB450
22 FORJ=1TOSKILL:NEXT
23 POKE36879,10
30 PRINT"[illegible]SCORE";SC
33 PRINT"[illegible]"TAB(12)"LIVES";N
45 X=PEEK(197)
47 P1=P:Q1=Q
49 IFX<>64ORST<>0THENSC=SC+1
50 IFST=1ORX=17THENP=P-22:Q=Q-22:GOTO70
52 IFST=2ORX=33THENP=P+22:Q=Q+22:GOTO70
54 IFST=8ORX=35THENP=P+1:Q=Q+1:GOTO70
56 IFST=4ORX=27THENP=P-1:Q=Q-1:GOTO70
58 IFST=9THENP=P-21:Q=Q-21:GOTO70
60 IFST=10THENP=P+23:Q=Q+23:GOTO70
62 IFST=6THENP=P+21:Q=Q+21:GOTO70
64 IFST=5THENP=P-23:Q=Q-23
70 IFPEEK(P)=34THENGOSUB175
72 IFX=43THENPOKE36879,136:FORJ=1TOSKILL:NEXT:POKE36879,10:SC=SC-100:IFSC<0THENN
=0
75 IFPEEK(P)=33ORPEEK(P)>41ANDPEEK(P)<48ORPEEK(P)=31THENP=P1:Q=Q1:GOTO135
80 IFPEEK(P)<40ANDPEEK(P)>35THEN140
120 POKEP1,32:POKEQ1,0
123 POKEP,35:POKEQ,5
125 IFN<1THEN800
135 GOTO30
140 REM HOME
141 POKEP1,32:POKEQ1,0
142 GOSUB850
143 POKE198,0
145 PRINT"[illegible]     WELL DONE      ";
147 GOSUB840
```

Program Listing 2 (cont.)

```
150 PRINT"     NOW TRY AGAIN    "
152 GOSUB840
155 PRINT"      HIT A KEY        "
157 GETL$:IFL$=""THEN157
158 PRINT"                          ";
160 SC=SC+L*50:L=L+1:M=M+5:SKILL=SKILL-SKILL/25
164 P=8152:Q=38872
165 POKE36879,136
170 PRINT"":GOTO17
175 FORJ=1TO7
176 POKEP1,32:POKEQ1,0
177 POKEP,34:POKEQ,J
180 POKE36878,15:POKE36876,128+J*10
185 FORK=1TO100:NEXTK
187 NEXTJ
190 N=N-1:POKE36878,0
195 POKEP,34:POKEQ,2:P=8152:Q=38872:RETURN
450 REM BORDER
455 PRINT"!!!!!!!!!!!!!!!!!!!!";:POKE7680,33:POKE3840
0,5
457 POKE8152,35:POKE38872,3
460 Z=7701:Y=38400-7680:FORJ=0TO22:POKEZ,33:POKEZ+Y,5:POKEZ+1,33:POKEZ+1+Y,5:Z=Z
+22:NEXT
465 PRINT"!!!!!!!!!!!!!!!!!!!!!!";
520 GOSUB700
530 REM TANK
540 PRINT"":PRINTTAB(INT(RND(1)*18)+1)"#%&'"
550 REM  MINES
555 FORJ=0TOM
560 D=INT(RND(1)*459)
565 IFPEEK(C+D)=32THENPOKEC+D,34:POKEC+D+Y,0:J=J+1
570 J=J-1:NEXT:RETURN
700 REM TREE
710 FORJ=0TO15
712 A=INT(RND(1)*415)+7703
720 IFPEEK(A)=32ANDPEEK(A+1)=32THENPOKEA,47:POKEA+1,46:POKEA+Y,5:POKEA+Y+1,5:GOT
O730
725 J=J-1:GOTO750
730 IFPEEK(A+22)=32ANDPEEK(A+23)=32THENPOKEA+22,45:POKEA+23,44:POKEA+22+Y,5:POKE
A+Y+23,5:GOTO740
735 J=J-1:POKEA,32:POKEA+1,32:GOTO750
740 IFPEEK(A+44)=32ANDPEEK(A+45)=32THENPOKEA+44,43:POKEA+45,42:POKEA+Y+44,5:POKE
A+Y+45,5:GOTO750
745 J=J-1:POKEA,32:POKEA+1,32:POKEA+22,32:POKEA+23,32
750 NEXTJ:RETURN
800 POKE36878,15
802 FORJ=20TO1STEP-1
804 POKE36876,255-J*5
806 NEXT:POKE36876,0
808 POKE36877,255
810 FORJ=1TO1000:NEXT
812 POKE36877,0
815 PRINT"      GAME OVER     "
816 GOSUB840
817 PRINT"  YOU SCORED";SC
818 GOSUB840
819 PRINT"  AND GOT TO LEVEL";L
820 GOSUB840:POKE198,0
825 PRINT"      HIT A KEY        "
829 GETK$:IFK$=""THEN829
830 CLR:RUN
840 FORJ=1TO1000:NEXT:RETURN
850 POKE36878,15:FORE=130TO254:POKE36876,E
855 NEXTE:POKE36878,0:POKE36876,0:RETURN
READY.
```





This month Runecaster picks up his joystick and grapples his way through arcade adventures.



THE ADVENTURE GAMES WE KNOW and love so much, provide us with a narrative as we move along — the descriptive location texts, that so often make or mar a good adventure. Additionally there may be a graphic representation of what we can see at the different locations.

But if we go back to the original basic scenario for an adventure, we may well find that there is another path by which we can achieve our goal — the arcade adventure. In certain cases it is difficult to know where to draw the line between what is purely an arcade game and what is an arcade adventure. Many of you will be familiar with Chuckie Egg (A 'n' F Software) and Manic Miner (Software Projects). Both of these involve moving around a specific series of locations collecting sufficient objects to enable you to move on to the next screen.

In neither of these two cases would anyone call them adventures. The puzzles involved are principally those of physical co-ordination and manipulation of the joystick, even though a logical path has to be determined as well.

As time passes, I think we will see more and more adventures which are joystick operated, and that have little or no text. That is not to say that they will take the place of the more traditional form but will appear as a separate branch of the expanding software options that become available.



Four arcade adventures

For some time there have been a number of programs for the Sinclair Spectrum that fit this category, such as Atic Atak and Sabre Wulf (both by Ultimate). Only recently have we had similar productions for the Commodore 64. Four programs that move along this alternative branch of the 'adventure tree' are: Hercules, Cuthbert Enters the Tombs of Doom, Quo Vadis and Impossible Mission. How do they rate as adventures?

Hercules by Interdisc, looks at first sight to be another Manic Miner variant. You play the part of Hercules and have to solve the twelve labours set him by King Eyrystheus. Each task is preceeded by a screenful of text describing the labour that has been set. There are no objects or treasures to be found as you progress through each adventure but what you *do* have to solve are a series of logical puzzles/mazes that will enable you to reach your objective.

Each labour consists of several screens (50 in all) but unlike Chuckie Egg or Manic Miner, not all of the floors are visible — you have to work out where they are. Some paths burst into flames when you tread on them, some disappear. Ropes that you jump for, could brek under your weight.

A reasonable amount of physical (joystick) dexterity is required and if you tarry too long at the beginning of each part, the floor beneath you burns and you perish. Be prepared to die fairly often as you determine the right path to success!

Unlike many related games, Hercules uses random access to the first eleven labours, so you are not continually faced with the same screen each time you start. Only having solved the first eleven can you then attempt the final part!

Although Hercules can only barely scrape into the 'arcade adventure' class, it nonetheless presents a time consuming and interesting game, with good graphics, and many of the problems that you will find in other types of adventure. With Cuthbert by Microdeal, not only do you get a game with the now familiar 'cuddly Cuthbert' as its hero, but also a very well-presented small booklet with "ye sette of instructions and clues". Read it carefully — not only will you gain an insight to playing the game, but Cuthbert's sense of humour should bring some amusement to the proceedings. I particularly liked the pages entitled 'Tombstones' — read the small print!

Your (Cuthbert's) task is to travel as far as possible through the ancient tombs of Ledromica. Travel far enough and you will come across areas of the tombs that represent letters; spot all of these and a prize awaits you from Microdeal. We are told that there are more than 200 chambers, so you have some way to go! Every so often you will find your way blocked by a locked door — easy enough to open, all you need is a key! The keys are heavy, so you may only hold one at a time and having used it, you must find another to open the next door. The air between

each set of locked doors is steadily being used up as you puff and pant your way around. To make it more difficult, there is a definite time limit, within which you must get a key to open the next door.

There are numerous treasures to be found along the way but do not let your greed overcome your need for oxygen! There are 'baddies' that appear shortly after you enter a chamber and you may destroy these by using the 'Ray of Ra' (but they still come back if you hang around). You also have another ultimate weapon that paralyses the evil ones — but having used it, you must then collect enough treasure from appropriately coloured rooms to top up its potency for re-use!

This is not a very intellectually demanding game — other than remembering where you last saw a key or spotting red herrings that may tempt you to use a key or your time unwisely. It can certainly become quite addictive up to a point and will tax the average adventurer's skill to progress past about 100 chambers! The only real disappointment is that although the graphics are quite good, they are repetitive and probably this fact alone will finally inhibit further incentive to continue.

Third in our present list of arcade adventures is Quo Vadis by The Edge. For those whose Latin is a little rusty, the title translates as "whither goest thou?" Very apt for an adventure game and especially this one whose sheer size of area to explore is probably approaching the equivalent of 1000 screens!

Your aim is to find the 'Sceptre of Hope' hidden deep underground — should you be the first to do so *and* send in a map of Quo Vadis, you may stand a chance of winning an actual sceptre worth £10000. Go to it!

You play the part of a 'spritely knight' whose name in another reincarnation must surely have been 'Spring Heeled Jack' — his capacity for jumping is more commendable! Not only is the movement very smooth but as he moves he fires a continuous stream of fire balls — very reassuring. There are many rocky platforms in the chambers for you to jump to/from, and you won't die if you fall from a great height — unless you happen to land in a pit of boiling tar! There are ropes to climb and a whole host of beasties to fight off. Should they or their missiles hit you, you lose 'strength points', which in turn may be accumulated (up to a maximum of 100) by finding a series of chests hidden in the caverns.

The background scenery is basically a variation of the same graphics again and again, but as the layout is so vast and the combinations so cleverly put together, you do not become bored with any similarity. Also the variety of 'baddies' keeps you on your toes.

There are a number of 'force fields' that these aggressive inhabitants are unable to pass and these enable our intrepid adventurer to nip out, take a few pot shots and retreat to safety. Repeating this routine allows you to clear the way with little or no loss to your strength. On the other hand there are some positions where it is impossible to employ this technique.

Although this game does not appear to provide a vast number of puzzles in the sense of the more conventional adventure, it certainly gives the would-be explorer a vast area to wander about, together with all the problems involved in mapping and finally solving such an epic.

Impossible Mission from Epyx must rate as the top of the range as far as arcade adventures go at the present time. Very briefly the scenario is that you (Special Agent 4125) must penetrate the underground stronghold of mad Professor Elvin Atombender (hereafter known as Elvin) and break his security codes to find his control centre.

Having located Elvin you must stop him completing his evil plans to destroy the world. Your predecessors, Agents 4116 and 4124 (may they rest in peace!) were able to send back a little information that may help you (all given in the excellent instruction booklet) but apart from this, your only weapons are your keen analytical mind and your MIA9366B pocket computer!

On loading the game you are welcomed by Prof Elvin with: "Another visitor; stay awhile. . .stay forever". This is to enable you to adjust your volume control. Yes, Impossible Mission has speech synthesis — clear as a bell too!

You start in a lift (elevator to our American cousins) and whilst here, or in one of the passages directly alongside, part of the screen displays your pocket computer output. With this you can map the rooms already visited, view the coded puzzle pieces you may have found and rotate these pieces to see if they fit a pattern or change their colour. Other coded patterns may be found to enable you to paralyse the robot guards or reset the moving floor panels. The PC display also keeps track of the time you have left to succeed in your mission.

Travelling in the lift allows access to other floors and rooms and moving along corridors or rooms creates excellent echoing footsteps. Press the fire button and Agent 4125 executes the most remarkable mid-air forward flip — very useful for somersaulting over robots (sometimes!) Enter a room and you hear Elvin's voice saying: "Destroy him my robots". You fall through a hole in the floor and hear yourself scream on the way down!

Impossible Mission is quite a remarkable game and, although I'm better at solving text puzzles than at solving jigsaws (the hidden coded patterns), I would recommend this one to anyone — if only to see what can be done with the modern home computer.

Each of the four 'adventures' I've reviewed requires a certain dexterity with the joystick — but if one has an *arcade* adventure, I think you must expect it, after all that is the 'raison d'être' of the arcade game.

Not all of these games will appeal to everyone, but try and get your local computer shop to get them up and running for you. I wouldn't mind betting that at least one of them will 'get you'. It will certainly prove interesting to see how this branch of the computer adventure saga progresses in the future.

More on mapping

Those that have read this column before will probably have realised that I put a great deal of stress on thorough mapping of advenuture games. In its most simple form, all you need is a large sheet of paper, a pencil (and an eraser for the odd mistake) and a cool head!

Where we can so easily go wrong is by not being neat and methodical and also the silly situation where we draw our boxes too large and don't have enough room on the paper, or make them too small to list all we find in that location!

Print n'Plotter Products have now come to the aid of the adventurer with their 'Adventure Planner'. In the past this firm have produced several extremely useful products that have made the life of the programmers much easier — mainly plotting sheets for Hi-Res pixel graphics, sprite design pads and so on.

Adventure Planner gives you 50 sheets of A3 size paper with 154 linked location boxes (on an 11 x 14 grid) printed in light grey. The boxes are a reasonable size to write a brief description of a location, together with what you have found there. Outline the box and the movement possibilities in pen or soft pencil and your map will stand out from the grid — simple, useful and logical — thanks P'n'PP.

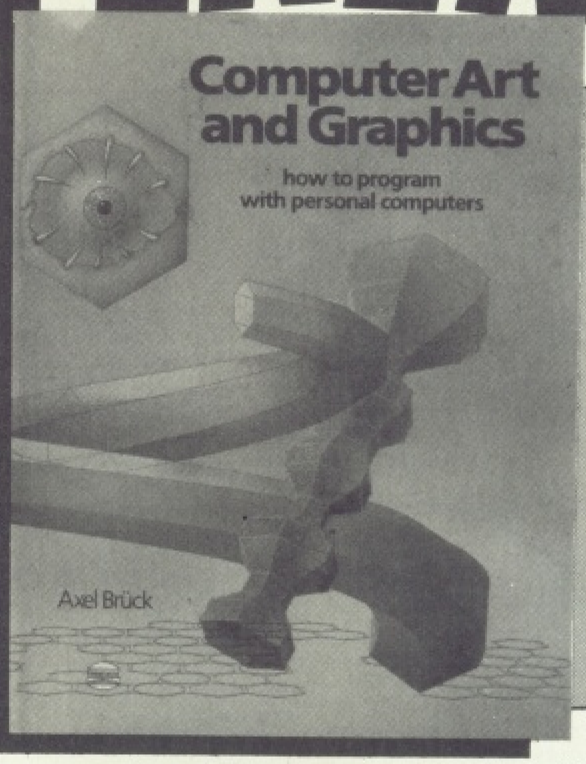
Notes on how to use, and a simple example are included on the first page — thereafter there is room for Adventure name, notes, dates, vocabulary etc. The only point I would add is; still do a rough initial map on a scrap of paper, as before serious mapping you should have some idea whether your adventure will develop to the North, South, East or West! For instance, Eric the Viking (Mosaic) starts in the East and all initial action takes place as you move West — therefore you would start mapping to the right of your page, wouldn't you?

Print n'Plotter Products A3 Adventure Planner should retail at £3.95 — if you cannot find one locally, write directly to them at: 19 Borough High Street, London SE1 9SE, and enclose £4.50 which includes post and packing.

REFERENCE

Fight off the mid-Winter blues by curling up in front of the fire with a book from this month's selection, reviewed by Allen Webb.

Title: Computer Art and Graphics
Author: Axel Bruck
Publisher: Petzold
Price: £14.95



WITH MOST OF THE HOME COMputers appearing today having some form of advanced graphics capability, interest is increasing in the area of computer art. This volume presents a serious but easy to use discussion of certain aspects of the topic. As I'll discuss later, the book is sadly lacking in a number of areas. The programs presented in the book were developed on the Apple computer, but can be readily extended to other machines.

Before continuing, I think it's a good idea to attempt to differentiate between graphics and art. The word "graphics" is often misused. My interpretation is that graphics are simply the representation of visual information — on your TV screen, a sheet of paper or a bathroom wall, for example, whereas art is the communication of ideas through a suitable medium such as stone, paint or your computer's graphics. The point the book and I are making is that your computer's graphics are simply a means to an end, and that end can be art.

In the foreword, the author of the book suggests two approaches to "computer art". First we have the computer specialist approach. In this case the computer is given a full set of design rules, a full set of algorithms and the result is a 100% computer generated product. On the other hand, the computer can be used as an aid (computer aided design-CAD) which generates a variety of shapes or effects. The picture is then finished by the artist giving a human element. The author adopts the latter approach, simply because art needs that human element.

To assist your creation of pictures, a library of routines are necessary. This book provides such tools. It has been known for a long time that three dimensional matter can be resolved into geometric forms. The cubists and Cezanne, for example, exploited this idea. Similarly, most of the routines given in the book produce hexagons in various forms. Using this simple shape, complex structure can be created.

The book is carefully structured and routines slowly become more complex as you progress. Things start with simple single and multiple shapes and progress to 2 dimensions, 3 dimensions, perspective and movement. The most complex routines enable you to build up shapes, ellipsoids and toroids using solid sub-shapes. Again these shapes can be moved and made to obey the rules of perspective. Throughout the book, colour plates of pictures are given showing what can be done. Most of these plates have to be drawn on a plotter with colour and other effects air brushed on afterwards.

The book is a joy to use and very easy to follow. The program listings are clearly laid out with copious notes in the text. The development of ideas is augmented by clear diagrams and the presentation is faultless. As a hard-back book, this volume is good value for money and deserves a place on your library shelf.

Inevitably, there are weaknesses. Firstly, the emphasis on hexagonal forms leads to rather monotonous pictures. Other forms such as triangles and tetrahedra have their own value, especially in the formation of smooth undulating surfaces. The author seems to have a predilection for surreal forms featuring eye balls. Whilst the pictures are technically excellent, more variety would be an idea. None of the routines use hidden line removal. Which can be slow and complex, but if you're using the TV rather than a plotter, it makes pictures easier to resolve. In spite of these reservations, the library of routines is extensive and powerful.

For the sake of 64 owners, the author includes a BASIC routine for plotting points which can be readily inserted into the drawing programs. Since all is in BASIC, the drawing of shapes tends to be slow. Owners of extended BASIC such as BC BASIC or Simon's BASIC should have little difficulty in converting the routines. I tested out the routines using Supersoft's GRAPHIX 64 and found the results quite acceptable.

The one drawback of the approach used in this book is that things are still rather mechanical. If you want to produce a freer form of art, you must resort to alternative methods. These include light pens, graphics pads and mechanical tracers. There are a number of excellent products about and the only limitation is your skill and imagination.

Title: Game Master
Author: P.K. McBride
Publisher: Longman
Price: £5.95

THIS BOOK INCLUDES LISTINGS FOR the reader to type in as well as programming advice. Whilst this means that you only get four games to play, you will learn something at the same time. The book is split into three sections covering action games, adventures and interactive or strategic games.

The author introduces each aspect of the type of game, demonstrates it with some example routines and then gives a full listing to type in.

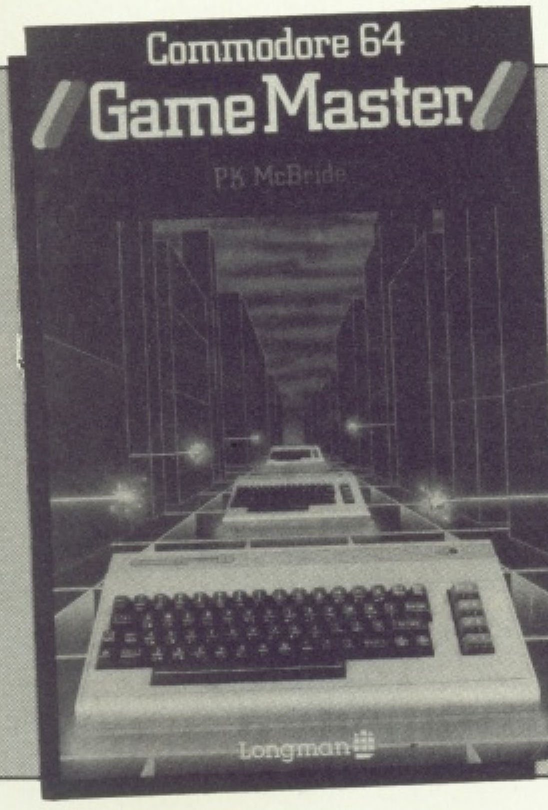
The first section deals with the creation of arcade type, "zap-the-alien" games. The areas discussed include the design of title pages, movement of sprites, delays, sights for shooting games, halls of fame, mazes, special effects and sound. There's quite a degree of overlap between sections but this tends to enhance the impact. Above all, there is sufficient information and ideas to stimulate most users. Much to my

surprise, the author acknowledges the speed limitations of BASIC and discusses the use of compilers and FORTH as faster alternatives.

As an adventure buff, I found the next section interesting. Again all aspects of the programming methods used are discussed, ranging from planning the scenario to randomness. The final section on strategy games is, not unsurprisingly, the shortest. General concepts are discussed, but when it comes down to it, simulations are pretty tough to program well. The book concludes with three appendices which discuss BASIC keywords, the design of characters and the manipulation of sprites.

Game Master is a rather rare breed. Firstly, it is fun in that it gives a number of full games and some short routines to input and play. Secondly, it's a mine of useful facts, hints and most importantly for games writers, ideas. At the price, it represents excellent value and is worth close examination.





Title: Artificial Intelligence on the Commodore 64
Author: K & S Brain
Publisher: Sunshine
Price: £6.95

ASK ANYONE IN THE STREET WHAT their conception of a computer is and they will probably think of something between Metal Mickey, Hal (from 2001) and R2D2. The sad truth is, however, that computers are basically pretty stupid and will only do what they are told. This book discusses some of the ways in which you can make your 64 appear to be intelligent. Rather than adopt an erudite, in-depth approach to the subject, the authors combine a little theory with demonstration programs which are described section by section. These programs can be used as a base upon which you can develop more complex models. To help the reader understand the techniques described, many flow diagrams are provided.

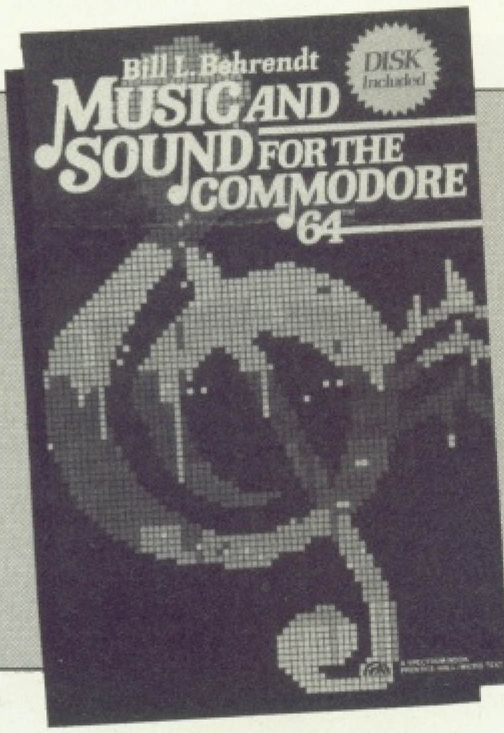
The first five chapters deal with the interpretation of language. These techniques feature in adventure games and such AI classics as Eliza and Abuse. The problem of parsing the input text, the recognition and interpretation of words and the formulation of responses are covered in detail. The chapters are carefully structured so that simple concepts are introduced and then refined to give greater flexibility and power.

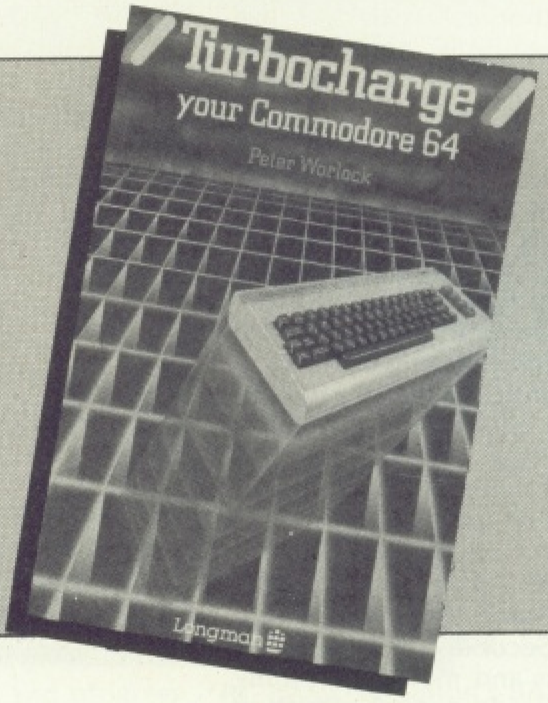
Having a computer that simply gives an amusing response to something you type in does become a little dull. In the next couple of chapters, the authors apply themselves to learning programs in which the computer acts as an expert with an in built data base. The idea is that the machine asks you questions on a subject and it remembers your replies, using them to modify its reactions later. One of the ideas developed gives you a fault finding system for cars. The computer asks you questions and gives it's diagnosis of the fault based on your responses.

The final chapters deal with the problem of recognition. Firstly, a technique called "Fuzzy Matching" is discussed. This is a system introduced in the USA to assist in census taking. By using a set of rules, names are reduced to a four character code. A program is provided which shows how the matching of names can be carried out using this technique. Finally, a simple demonstration of shape recognition is given.

This is both an educational and fun book which handles a potentially complex topic in a nice simple manner. If you want to really see what your 64 can do rather than just play games, try this book, it's well worth reading.

REFERENCE LIBRARY





Title: Music and Sound for the Commodore 64
Author: B.L. Behrendt.
Publisher: Prentice-Hall
Price: £34.78

THE INCLUSION OF SOPHISTICATED sound chips such as the 6581 means that you have the added extra of sound synthesis without the need to buy a separate synthesiser. In addition to the annoying lack of sound commands on the 64, sound synthesis is a complex field that isn't readily mastered. This book is one of many dedicated to the discussion of sound synthesis. The package in fact, comprises a disc and a book. Rather than force you to type in all the programs discussed, they are recorded on the disc leaving you free to concentrate on the theory of the music.

The book starts with three chapters on theory. These cover the physics of sound and the fundamental units of sound synthesisers. These chapters are important since they introduce the subject and describe how the operations encountered later work. Details of the various registers in the 6581 are given along with operational parameters.

The remaining chapters cover many interesting aspects of sound synthesis and provide routines to demonstrate the lessons learned. The most noteworthy routines provide facilities for the development of sounds, music editing and a simple sequencer. The information in the text provides example settings which can be used in the programs to produce interesting effects.

Computers can be programmed with certain rules of operation to behave in a human-like manner. This is often called artificial intelligence. A similar exercise can be carried out with sound so that the computer generates music. Simple music generation techniques are introduced into the book demonstrating how sound based on simple scales or chords can be generated. Surprisingly the effects are quite pleasant if not a little monotonous. Finally, you are provided with a number of routines for sound effects which are not only suitable for use in your own programs but can be developed to give further effects.

Overall, this is a highly entertaining and useful package which is well written and presented but the price is ludicrous: half the price would be nearer the mark.

Title: Turbocharge your 64
Author: P. Worlock
Publisher: Longman
Price: £5.95

AUTHORS OF BOOKS LIKE THIS OWE A lot to Commodore's policy of providing useless manuals with their computers. This book, rather than providing information on the basics of the 64, claims to tell you how "the Professionals" do it. I rather dispute this claim but some readers might believe it. In essence the book gives masses of hints, tricks and wheezes which will certainly embellish your masterpieces. Because of this approach, the treatment of information tends to be of a "potted" nature rather than "in depth".

To get you in the mood for writing wonder programs, we start with program structure and a discussion of the various functions available from BASIC. As a simple intoduction to structure, these chapters aren't bad, but the lessons learnt aren't very earth shattering. The section on functions discusses the transcendentials and RND but doesn't cover any new ground.

One of the "in" phrases in computer circles is "user friendly". The author recognises this attribute and gives quite a decent section on interaction. This covers methods of inputting information, error trapping and formatting of displays. A short machine code PRINT AT routine is given as a handy utility. The area of interfacing is extended later in the book to cover joysticks. Again a machine code utility is supplied as an aid.

The sections on graphics are of a higher standard and cover most of the important areas. There's a fair discussion of the memory organization of the 64 and how to redefine characters, the use of sprites and high resolution graphics. Simple character movement is described as is the use of sprites to give simple animation effects.

Finally for lovers of music and sound effects, the operation of the SID is examined along with a reasonable amount of musical theory.

I found it a little difficult to decide whether I loved or loathed this book. The treatment of the material is good but the book does not give the claimed "professional' techniques. The material supplied is quite standard and no more than I would expect in a manual. Throughout there are short BASIC routines which demonstrate various points and the presentation is lively and attractive. If I do have a complaint, it is that the chapters tend to be rather short and occasionally superficial.

CHERRY PICKER

Collect cherries as you fend off the spiders in this fruity, fun-filled game from F.G. Tout.



YOUR TASK IS TO GUIDE Freddie through 5 different waves, collecting cherries as you go. You score 100 points for each cherry you collect. But, watch out for the spiders, etc...they're out to stop you.

Plug your joystick into port 2; left=left, right=right, fire button to jump. You can guide Freddie after he has jumped: for a long jump, keep the joystick to the left or right; for a short jump, release the joystick; to jump up, do not move the joystick at all.

You score 100 points for each cherry you collect. You have 3 lives with an extra life on waves 3 and 5.

Program Information

Part 1 ... This is all sprite data.

Part 2

10	299	U.D.G.s
300	430	New Letters
999	2040	Music Interrupt
2100	2380	Music Data
19999	....	Machine Code (REMmed)

Part 3

2	81	Subroutines to check for collisions with spiders or other nasties, and whether you pick a cherry.
100	400	Variables and define-functions.
500	550	Main frame
900	1059	Screen 1
1100	1172	Screen 2
1200	1295	Screen 3
1300	1395	Screen 4
1400	1498	Screen 5
40000	41111	Keep Score etc.
49999	50030	Game Over
60000	60299	Title page

Variables

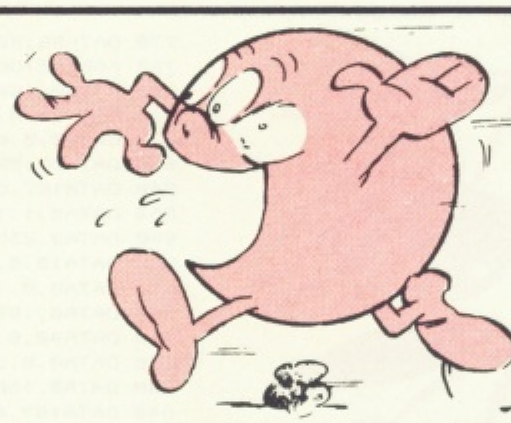
V	= Sprite variable
Li	= Lives
Le	= Level
S1,S2,S3	= Sound
J	= Position of Fred
G	= Var
L	= J's port
M1,M2,M3	= M/C addresses
H,K	= Collision registers

Program Listing

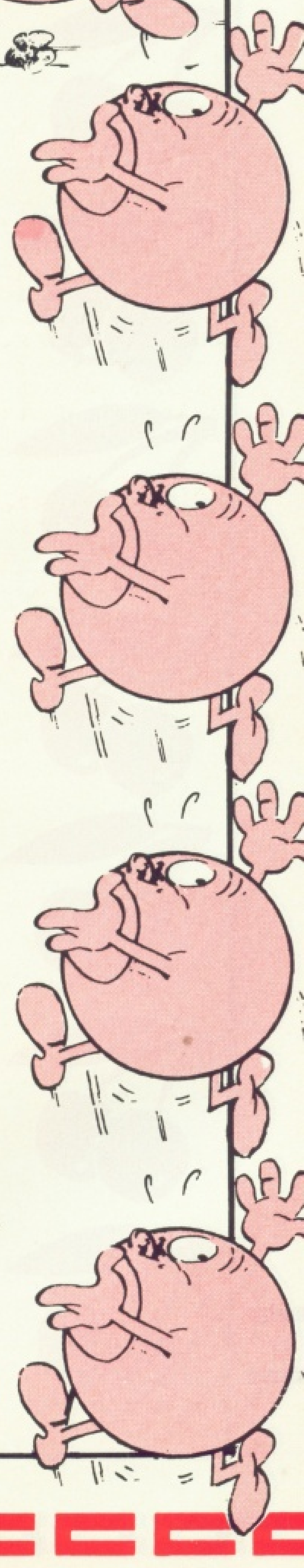
```
4 REM CLR-ORN
5 POKE53280,9:POKE53281,9:PRINT"■■CHERRY PICKER BY FRANK TOUT"
10 I=0:FORI=0TO64*7-1:READA:POKE210*64+I,A:NEXT
20 DATA2,168,0,10,170,0,10
25 DATA82,0,9,32,0,9,84
30 DATA0,9,80,0,0,64,0
32 DATA2,96,0,10,168,0,9
35 DATA168,0,9,168,0,9,168
40 DATA0,9,168,0,13,252,0
45 DATA13,251,0,3,255,0,95
50 DATA207,192,126,3,192,112,0
55 DATA112,64,0,80,0,0,84,0
60 DATA2,168,0,10,170,0,10
65 DATA82,0,9,32,0,9,84
```

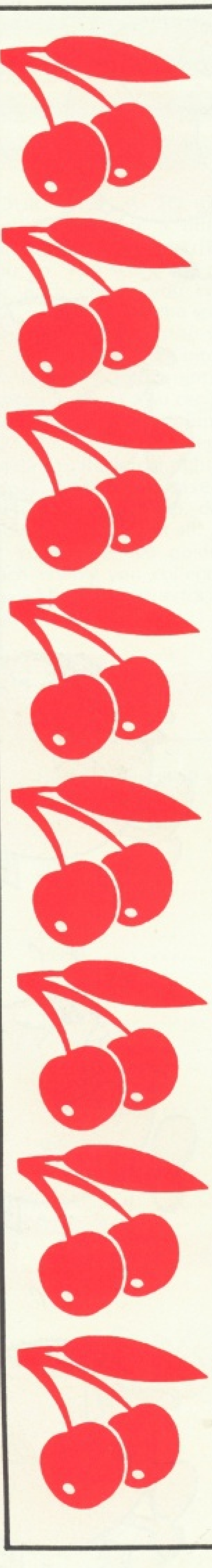




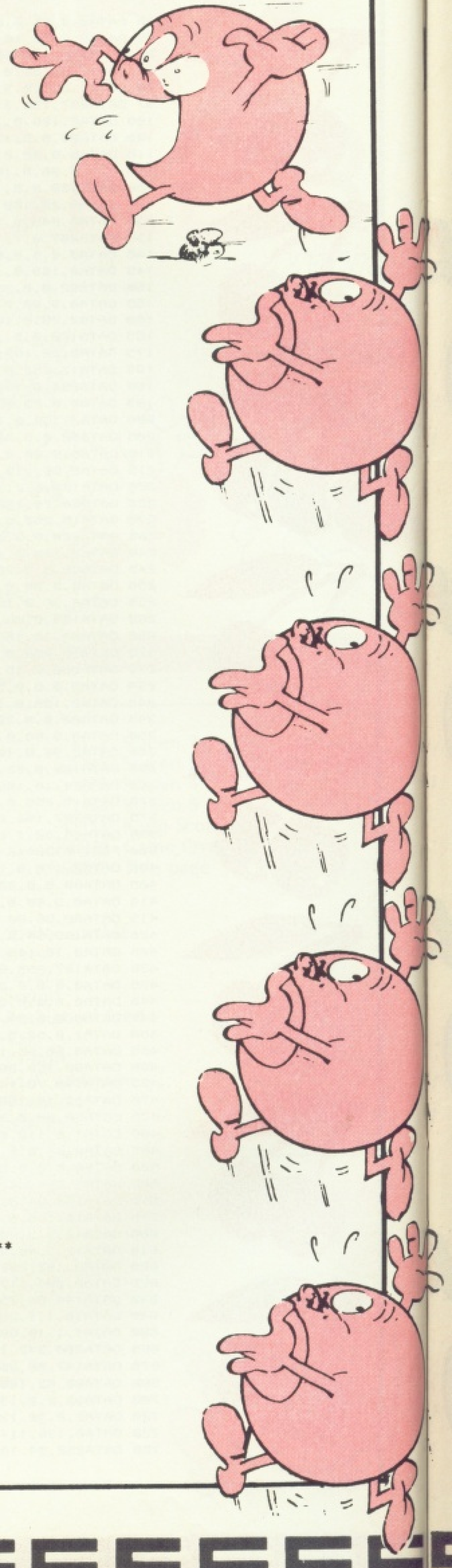


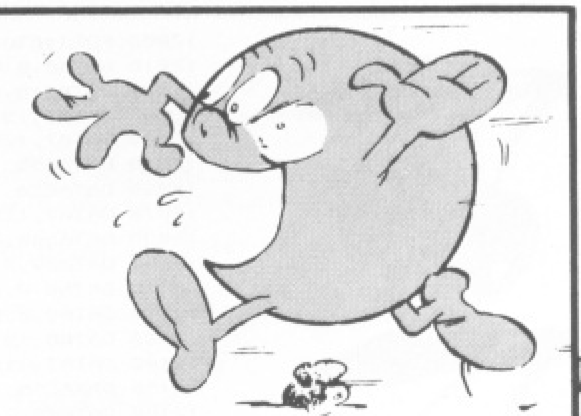
```
70 DATA0,9,80,0,0,64,0
75 DATA2,96,0,10,168,0,9
80 DATA168,0,9,168,0,9,168
85 DATA0,5,168,0,7,252,0
90 DATA7,251,0,3,255,0,23
97 DATA207,192,31,195,192,31,1,192,16,1,64,0,1,80,0
100 DATA2,168,0,10,170,0,10
105 DATA82,0,9,32,0,9,84
110 DATA0,9,80,0,0,64,0
115 DATA2,96,0,10,168,0,9
120 DATA168,0,5,168,0,6,168
125 DATA0,22,169,0,31,253,0
130 DATA0,248,0,7,255,0,7
135 DATA207,0,7,15,0,4,7
140 DATA0,0,5,0,0,5,64,0
145 DATA2,168,0,10,170,0,10
150 DATA82,0,9,32,0,9,84
155 DATA0,9,80,0,0,64,0
160 DATA2,96,0,10,168,,9
165 DATA168,0,5,168,0,6,168
175 DATA0,22,169,0,31,253,0
185 DATA15,252,0,1,124,0,1
190 DATA252,0,1,204,0,1,52
195 DATA0,0,20,0,0,21,0,0
200 DATA2,168,0,10,170,0,10
205 DATA82,0,9,32,0,9,84
210 DATA0,9,80,0,0,64,0
215 DATA2,96,,10,168,,21
220 DATA168,0,21,169,0,90,169
225 DATA64,74,168,64,15,252,0
230 DATA15,252,0,15,220,0,3
235 DATA220,0,3,208,0,3,208,0,1,192,0,1,80,0,0
240 DATA2,168,0,10,170,0,10
245 DATA82,0,9,32,0,9,84
250 DATA0,9,80,0,0,64,0
255 DATA2,96,0,10,168,0,21
260 DATA169,0,85,169,64,74,168
265 DATA64,10,168,0,15,252,0
270 DATA15,252,0,15,252,0,3
275 DATA252,0,15,212,0,7,20
280 DATA0,5,5,0,5,64,0,0
340 DATA2,168,0,10,170,0,10
345 DATA82,0,9,32,0,9,84
350 DATA0,9,80,0,0,64,0
355 DATA2,96,0,10,168,0,21
360 DATA169,0,85,169,64,74,168
365 DATA64,10,168,0,15,252,0
370 DATA15,255,0,15,255,0,63
375 DATA207,192,255,1,192,92,1
380 DATA64,80,1,80,20,0,0,0
390 FORI=0TO64*6-1:READA:POKE225*64+I,A:NEXT
400 DATA2,170,0,10,168,0,10
405 DATA80,0,9,32,0,9,84
410 DATA0,9,80,0,64,64,4
415 DATA82,96,20,22,168,80,5
420 DATA169,64,9,169,0,10,168
425 DATA0,10,168,0,63,255,1
430 DATA127,255,245,124,3,244,64
435 DATA0,0,0,0,0,0,0
440 DATA0,0,0,0,0,0,0,0
445 DATA0,0,0,84,0,0,20
450 DATA1,0,52,5,0,60,20
455 DATA0,60,80,168,62,160,168
460 DATA62,168,86,14,168,86,14
465 DATA169,70,14,168,86,14,168
470 DATA86,14,160,24,6,8,24
475 DATA60,20,0,252,5,0,240
480 DATA1,0,112,0,0,64,0
485 DATA0,64,0,0,80,0,0,0
500 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
505 DATA3,244,64,255,245,124,255,1,127,168,0,63,168,0,10,169,0,10
510 DATA169,64,9,168,80,5,96,20,22,64,4,82,9,0,9,84,0,9,32,0,10,80,0
515 DATA10,168,0,2,170,0,0
600 DATA48,3,3,204,12,3,204
610 DATA12,3,48,3,0,3,48,3,3,48,204,3,48,204
620 DATA0,192,252,84,0,204,80
625 DATA0,204,117,64,0,245,0,16,247,17,1,252,16,0,252,20,17,254,145,24,254
630 DATA144,86,254,152,70,254,169,86,62,168,86,62,160,40,0
640 DATA16,1,1,68,4,1,68,4,1,16,1,0,1,16
660 DATA1,1,16,68,1,16,68,0,64,84,84,0,68,80,0,68,117,64,0,245,0
665 DATA204,247,16,0,252,28,3,252,16,16,254,156,28,254
670 DATA147,86,254,152,118,254,169
680 DATA86,62,168,86,62,160,40,0
700 DATA32,2,2,136,8,2,136,8,2,32,2,0,2,32
720 DATA2,2,32,136,2,32,136,0,128,168,84,0,136,80
730 DATA0,136,117,64,0,245,0,32,247,18,8,252,16,0
750 DATA252,24,18,254,144,16,254,146,90,254,153,150,254,169
```





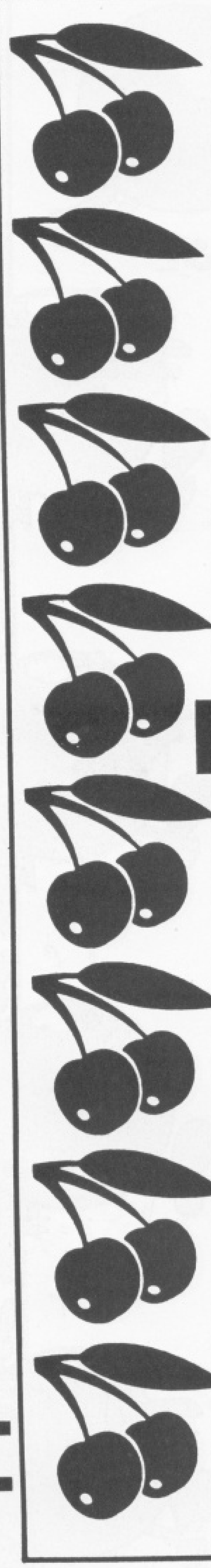
```
770 DATA86,62,168,86,62,160,40,0
790 FORI=0TO64*8-1:READA:POKE217*64+I,A:NEXT
800 DATA0,168,0,2,170,0,10
810 DATA90,0,0,70,0,1,86
820 DATA0,0,84,0,0,16,0
830 DATA0,152,0,2,170,0,2
840 DATA167,0,2,150,0,2,90
850 DATA0,1,160,0,1,170,0
860 DATA3,255,0,3,207,0,15
870 DATA15,0,52,12,0,80,15
880 DATA0,0,13,0,0,21,0,0
900 DATA0,168,0,2,170,0,10
910 DATA90,0,0,70,0,1,86
920 DATA0,0,84,0,0,16,0
930 DATA0,152,0,2,170,0,2
940 DATA167,0,2,166,0,2,150
950 DATA0,2,90,0,2,106,0
960 DATA3,255,0,3,255,0,15
970 DATA207,0,63,15,192,20,3
980 DATA192,80,1,64,0,5,64,0
1000 DATA0,168,0,2,170,0,10
1010 DATA90,0,0,70,0,1,86
1020 DATA0,0,84,0,0,16,0
1030 DATA0,152,0,2,170,0,2
1040 DATA166,0,2,166,0,2,166
1050 DATA0,2,150,0,2,154,0
1060 DATA3,223,0,3,255,0,3
1070 DATA207,192,15,195,240,15,0
1080 DATA240,52,0,80,84,1,80,0
1100 DATA0,168,0,2,170,0,10
1110 DATA90,0,0,70,0,1,86
1120 DATA0,0,84,0,0,16,0
1130 DATA0,152,0,2,170,0,2
1140 DATA166,0,2,166,0,2,166
1150 DATA0,2,166,0,2,166,0
1160 DATA3,247,0,3,255,192,3
1170 DATA195,240,3,0,252,3,64
1180 DATA52,3,64,4,5,64,20,0
1200 DATA0,168,0,2,170,0,10
1210 DATA90,0,0,70,0,1,86
1220 DATA0,0,84,0,0,16,0
1230 DATA0,152,0,2,170,0,2
1240 DATA166,0,2,166,0,2,166
1250 DATA0,2,166,0,2,166,0
1260 DATA3,247,0,3,255,80,3
1270 DATA207,208,0,243,208,0,240
1280 DATA16,0,80,16,1,80,0,0
1300 DATA0,168,0,2,170,0,10
1310 DATA90,0,0,70,0,1,86
1320 DATA0,0,84,0,0,16,0
1330 DATA0,152,0,2,170,0,2
1340 DATA166,0,2,166,0,2,166
1350 DATA0,2,165,0,2,169,0
1360 DATA3,255,0,3,245,0,3
1370 DATA253,0,0,205,0,0,253
1380 DATA0,0,212,0,0,84,0,0
1400 DATA0,168,0,2,170,0,10
1410 DATA90,0,0,70,0,1,86
1420 DATA0,0,84,0,0,16,0
1430 DATA0,152,0,2,170,0,2
1440 DATA166,0,2,166,0,2,166
1450 DATA0,2,165,0,2,169,0
1460 DATA3,255,0,3,252,0,3
1470 DATA220,0,0,92,0,1,79
1480 DATA0,0,13,0,0,21,0,0
1500 DATA0,168,0,2,170,0,10
1510 DATA90,0,0,70,0,1,86
1520 DATA0,0,84,0,0,16,0
1530 DATA0,152,0,2,170,0,2
1540 DATA166,0,2,166,0,2,166
1550 DATA0,6,169,64,6,170,64
1560 DATA3,255,0,3,255,0,15
1570 DATA207,192,13,3,240,21,0
1580 DATA240,0,0,208,0,1,80,0
9000 FORI=0TO127:READA:POKE231*64+I,A:NEXT:REM**SP**
9001 DATA7,1,192,3,131,128,28
9002 DATA198,48,54,108,216,99,109
9003 DATA140,193,239,6,192,252,6
9004 DATA192,124,6,224,254,14,1
9005 DATA239,0,31,171,240,48,214
9006 DATA24,96,124,12,192,198,6
9007 DATA193,131,6,227,1,142,6
9008 DATA0,192,6,238,192,7,131
9009 DATA192,3,1,128,1,199,0,0
9011 DATA0,16,0,0,56,0,0
9012 DATA124,0,0,108,0,28,108
9013 DATA112,62,108,248,99,253,140
9014 DATA193,255,6,192,254,6,193
```





```
9015 DATA239,6,231,255,238,12,254
9016 DATA48,24,124,24,48,198,12
9017 DATA96,198,6,192,254,6,192
9018 DATA108,6,224,56,14,0,0
9019 DATA0,0,0,0,0,0,0,0
10000 FORI=0TO191:READA:POKE233*64+I,A:NEXT
10010 DATA0,24,0,0,60,0,192
10012 DATA90,3,192,126,3,224,102
10013 DATA7,112,102,14,121,189,158
10014 DATA61,255,188,63,255,252,63
10016 DATA255,252,55,247,204,55,247
10018 DATA236,7,115,224,14,123,240
10020 DATA14,249,240,12,253,240,29
10022 DATA255,184,29,255,216,31,255
10024 DATA248,51,207,60,33,134,28,0
10030 DATA0,24,0,0,60,0,192
10032 DATA90,0,192,126,3,224,126
10033 DATA7,240,102,15,121,189,158
10034 DATA125,255,190,63,255,252,63
10036 DATA255,252,55,247,204,55,247
10038 DATA236,55,115,236,14,123,240
10040 DATA14,249,240,12,253,240,29
10042 DATA255,184,29,255,216,31,255
10044 DATA248,63,158,120,51,12,48,0
10050 DATA0,24,0,0,60,0,192
10052 DATA90,0,0,126,0,192,126
10053 DATA3,224,126,7,241,189,143
10054 DATA125,255,190,127,255,254,63
10056 DATA255,252,51,247,204,55,247
10058 DATA236,55,127,236,62,127,252
10060 DATA15,253,240,13,253,240,29
10062 DATA255,248,29,255,216,31,255
10064 DATA248,62,121,228,24,48,196,0
11000 FORI=0TO64*6-1:READA:POKE237*64+I,A:NEXT
11010 DATA0,0,0,0,0,0,255
11012 DATA255,254,0,56,0,15,255
11014 DATA224,12;0,96,108,40,96
11016 DATA204,146,96,204,16,96,205
11018 DATA17,96,252,16,124,205,17
11020 DATA108,204,0,108,204,130,111
11022 DATA108,40,96,12,0,96,15
11024 DATA255,224,0,56,0,0,56
11026 DATA0,0,56,0,0,254,0,0
11110 DATA0,0,0,0,0,0,63
11112 DATA255,248,0;56,0,15,255
11114 DATA224,12,0,96,12,40,96
11116 DATA108,130,96,204,2,96,205
11118 DATA5,96,252,8,124,205,17
11120 DATA108,204,0,111,108,130,96
11122 DATA12,40,96,12,0,96,15
11124 DATA255,224,0,56,0,0,56
11126 DATA0,0,254,0,0,0,0,0
11210 DATA0,0,0,0,0,0,15
11212 DATA255,224,0,56,0,15,255
11214 DATA224,12,0,96,12,40,96
11216 DATA12,130,96,108,0,96,205
11218 DATA1,96,252,0,124,205,31
11220 DATA111,108,0,96,12,130,96
11222 DATA12,40,96,12,0,96,15
11224 DATA255,224,0,56,0,0,254
11226 DATA0,0,0,0,0,0,0,0
11310 DATA0,0,0,0,0,0,3
11312 DATA255,128,0,56,0,15,255
11314 DATA224,12,0,96,12,40,96
11316 DATA12,130,96,12,0,96,109
11318 DATA1,96,252,16,127,109,25
11320 DATA96,12,8,96,12,134,96
11322 DATA12,40,96,12,0,96,15
11324 DATA255,224,0,254,0,0,0
11326 DATA0,0,0,0,0,0,0,0
11410 DATA0,0,0,0,0,0,0
11412 DATA254,0,0,56,0,15,255
11414 DATA224,12,0,96,12,40,96
11416 DATA12,130,96,12,0,96,13
11418 DATA1,111,252,16,124,13,17
11420 DATA96,12,16,96,12,146,96
11422 DATA12,56,96,12,0,96,15
11424 DATA255,224,0,56,0,0,254
11426 DATA0,0,0,0,0,0,0,0
11510 DATA0,0,0,0,0,0,0
11512 DATA56,0,0,56,0,15,255
11514 DATA224,12,0,96,12,40,96
11516 DATA12,130,96,76,0,111,205
11518 DATA1,108,252,8,124,205,17
11520 DATA96,76,32,96,12,194,96
11522 DATA12,40,96,12,0,96,15
11524 DATA255,224,0,56,0,0,56
11526 DATA0,0,254,0,0,0,0,0
```

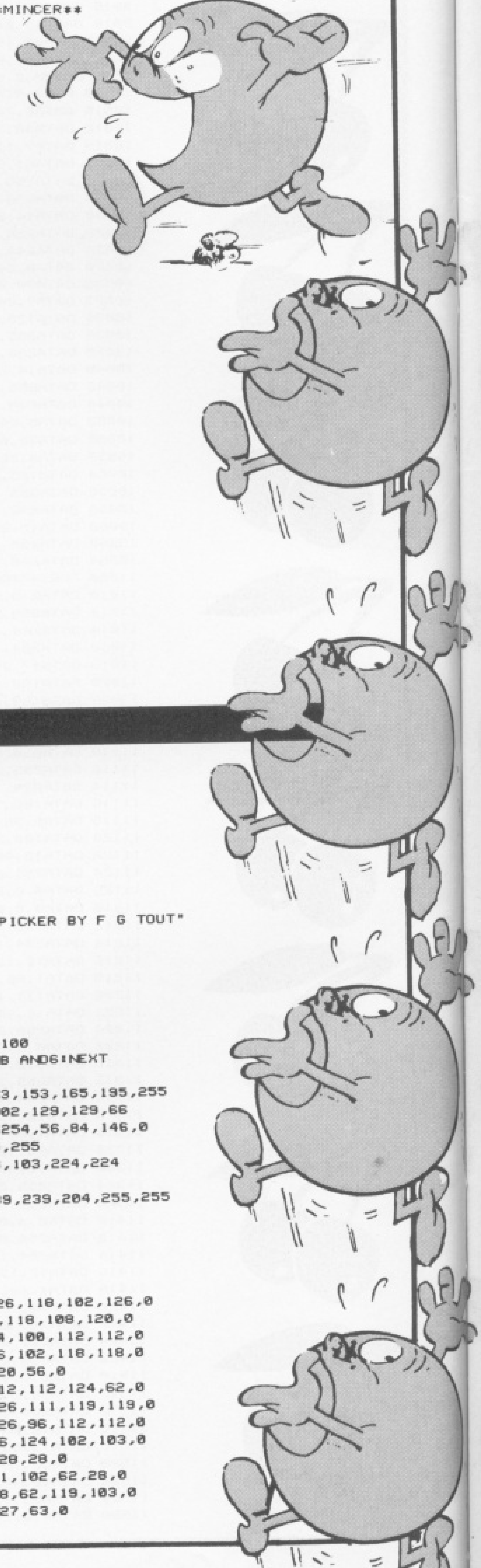




```
12000 FORI=0TO64*4-1:READA:POKE243*64+I,A:NEXT:REM**MINCER**
12010 DATA0,0,0,0,0,0,0
12020 DATA0,0,63,240,0,31,224
12030 DATA0,15,192,0,7,128,0
12040 DATA7,128,0,7,134,0,63
12050 DATA255,192,55,254,176,247,254
12060 DATA220,7,133,114,7,128,92
12070 DATA7,128,49,7,128,26,63
12080 DATA240,84,60,96,25,0,96
12090 DATA28,0,96,62,0,240,126,0
12110 DATA0,0,0,0,0,0,0
12120 DATA0,0,63,240,0,31,224
12130 DATA0,15,192,0,7,128,0
12140 DATA7,128,0,7,134,0,63
12150 DATA255,192,247,254,160,7,254
12160 DATA88,7,133,80,7,128,108
12170 DATA7,128,41,7,128,20,63
12180 DATA240,84,24,96,19,60,96
12190 DATA29,0,240,60,0,0,126,0
12210 DATA0,0,0,0,0,0,0
12220 DATA0,0,63,240,0,31,224
12230 DATA0,15,192,0,7,128,0
12240 DATA7,128,0,7,134,0,255
12250 DATA255,136,7,254,160,7,254
12260 DATA216,7,133,80,7,128,110
12270 DATA7,128,43,7,128,21,63
12280 DATA240,94,24,96,27,24,240
12290 DATA29,60,0,54,0,0,111,0
12310 DATA0,0,0,0,0,0,0
12320 DATA0,0,63,240,0,31,224
12330 DATA0,15,192,0,7,128,0
12340 DATA7,128,0,247,134,0,63
12350 DATA255,0,7,254,96,7,254
12360 DATA216,7,133,20,7,128,76
12370 DATA7,128,47,7,128,21,63
12380 DATA240,86,24,240,10,24,0
12390 DATA29,24,0,30,60,0,125,0
13000 POKE198,2:POKE631,13:LOAD
READY.
```

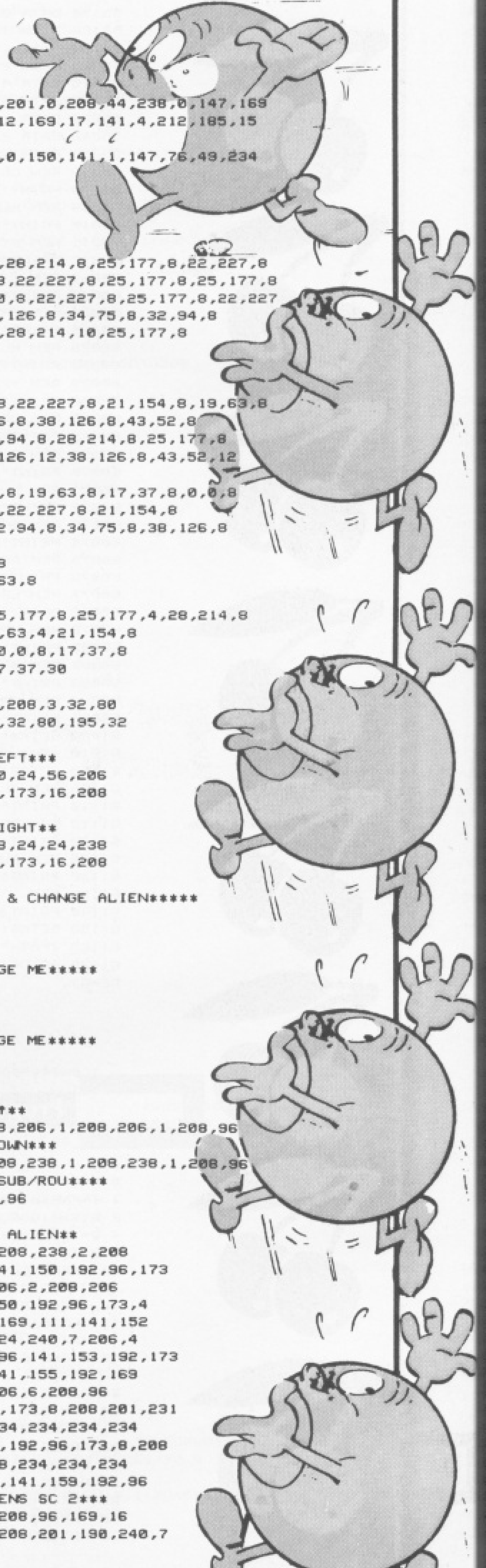
Program Listing Part 2

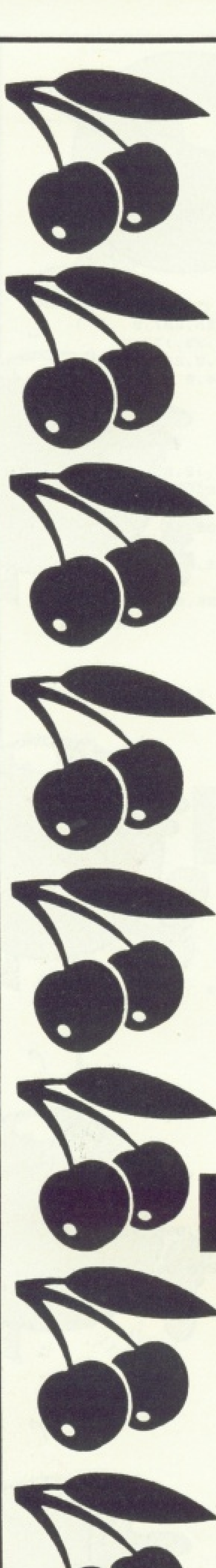
```
0 REM*****************************************
1 REM**              CHERRY PICKER          **
2 REM**               BY F G TOUT           **
3 REM** THIS PART OF THE PROGRAM IS ALMOST **
4 REM** ALL MACHINE CODE AND CARE MUST BE   **
5 REM** TAKEN ON ENTERING IT.
6 REM*****************************************
9 REM CLR BLK
10 PRINT"┘":POKE53280,2:POKE53281,2:PRINT"■CHERRY PICKER BY F G TOUT"
15 REM***DOWNLOAD UDGS**
20 POKE56334,PEEK(56334)AND254
30 POKE1,PEEK(1)AND251
40 FORG=0TO64*8:POKE12288+G,PEEK(53248+G):NEXT
50 POKE1,PEEK(1)OR4
60 POKE56334,PEEK(56334)OR1
100 READA%:T=T+1:IFA%<>-1THENPOKE12288+G+T,A%:GOTO100
101 FORI=0TO60*8STEP8:A=12288+1+I:B=PEEK(A):POKEA,B AND6:NEXT
102 DATA0,0,0,0,0,0
105 DATA239,239,0,219,219,0,249,249,255,195,165,153,153,165,195,255
110 DATA224,48,24,60,110,118,60,0,90,189,126,153,102,129,129,66
115 DATA129,255,129,255,129,255,129,255,146,84,56,254,56,84,146,0
120 DATA24,28,88,56,26,28,24,24,0,0,0,0,60,255,255,255
125 DATA255,126,124,60,56,24,24,16,7,31,115,99,103,103,224,224
130 DATA255,255,255,255,255,255,255,255
135 DATA7,31,59,251,251,239,255,255,224,248,236,239,239,204,255,255
140 DATA255,255,239,239,204,255,255,255
299 DATA-1
300 FORI=0TO215:READA:POKE12288+I,A:NEXT
302 REM**LETTERS**
305 DATA0,0,0,0,0,0,0,0
310 DATA48,124,108,126,118,102,102,0,48,124,108,126,118,102,126,0
320 DATA60,108,108,96,96,118,62,0,120,108,118,118,118,108,120,0
330 DATA126,118,96,124,96,118,126,0,126,118,96,124,100,112,112,0
340 DATA60,118,96,110,118,126,60,0,108,110,118,126,102,118,118,0
350 DATA124,56,24,28,28,30,62,0,62,30,28,12,108,120,56,0
360 DATA110,108,124,112,124,108,110,0,96,96,112,112,112,124,62,0
370 DATA54,119,127,107,119,119,119,0,54,118,122,126,111,119,119,0
380 DATA62,103,115,119,103,115,62,0,126,103,115,126,96,112,112,0
390 DATA62,103,115,115,115,62,15,0,126,115,103,126,124,102,103,0
400 DATA62,115,97,62,67,103,62,0,127,60,24,24,28,28,28,0
410 DATA103,103,71,103,102,126,60,0,103,103,103,71,102,62,28,0
420 DATA119,119,107,107,127,119,54,0,103,119,62,28,62,119,103,0
430 DATA115,51,62,24,24,28,28,0,126,15,12,24,48,127,63,0
999 REM***MUSIC INTERRUPT***
```





```
1000 FORT=0TO2:FORX=0TO255:READA:IFA=-1THEN1040
1010 POKE36864+T*256+X,A:NEXTX
1040 NEXTT
1050 DATA169,146,141,21,3,169,0,141,20,3,96,-1
1060 DATA169,234,141,21,3,169,49,141,20,3,96,-1
1070 DATA172,0,147,192,17,240,57,206,1,147,173,1,147,201,0,208,44,238,0,147,169
1080 DATA0,141,4,212,169,14,141,5,212,169,15,141,6,212,169,17,141,4,212,185,15
1090 DATA167
1095 DATA185,0,148,141,1,212,185,0,149,141,0,212,185,0,150,141,1,147,76,49,234
1097 DATA169,0,141,0,147,141,4,212,24,144,242,-1
2000 POKE37632,0:POKE37633,0
2010 FORT=0TO155:READHF,LF,DU
2020 POKE37888+T,HF:POKE38144+T,LF:POKE38400+T,DU
2040 NEXTT:POKE37380,T+1
2100 DATA21,154,8,21,154,8,25,177,8,25,177,8,32,94,8,28,214,8,25,177,8,22,227,8
2110 DATA22,227,8,25,177,8,22,227,8,21,154,8,21,154,8,22,227,8,25,177,8,25,177,8
2120 DATA32,94,8,32,94,8,34,75,8,32,94,8,28,214,8,0,0,8,22,227,8,25,177,8,22,227
2130 DATA8,21,154,8,19,63,8,19,63,8,17,37,8,0,0,8,38,126,8,34,75,8,32,94,8
2140 DATA28,214,8,34,75,8,32,94,10,28,214,8,25,177,8,28,214,10,25,177,8
2150 DATA22,227,8,21,154,8,22,227,8
2160 DATA25,177,8,28,214,8,34,75,8,32,94,8,28,214,8
2170 DATA25,177,8,32,94,8
2180 DATA28,214,8,25,177,8,22,227,8,28,214,8,25,177,8,22,227,8,21,154,8,19,63,8
2200 DATA19,63,8,17,37,8,0,0,8,34,75,8,34,75,8,38,126,8,38,126,8,43,52,8
2210 DATA38,126,8,34,75,8,32,94,8,34,75,8,34,75,8,32,94,8,28,214,8,25,177,8
2220 DATA28,214,8,28,214,8,0,0,8,34,75,8,34,75,8,38,126,12,38,126,8,43,52,12
2230 DATA38,126,8,34,75,8,32,94,8,25,177,8
2240 DATA25,177,8,22,227,8,21,154,12,19,63,12,21,154,8,19,63,8,17,37,8,0,0,8
2250 DATA28,214,8,28,214,8,32,94,8,28,214,8,25,177,8,22,227,8,21,154,8
2300 DATA21,154,8,22,227,8,0,0,8,28,214,8,28,214,8,32,94,8,34,75,8,38,126,8
2310 DATA34,75,8,43,52,8,38,126,8,0,0,8,25,177,8
2320 DATA25,177,8,25,177,8,22,227,8,21,154,8,22,227,8
2330 DATA21,154,8,17,37,8,0,0,8,17,37,8,21,154,8,19,63,8
2340 DATA22,227,8,25,177,8,28,214,8
2350 DATA28,214,8,25,177,8,0,0,8,22,227,8,22,227,4,25,177,8,25,177,4,28,214,8
2360 DATA25,177,8,22,227,8,21,154,8,0,0,8,19,63,8,19,63,4,21,154,8
2370 DATA21,154,4,22,227,8,21,154,8,19,63,8,17,37,8,0,0,8,17,37,8
2380 DATA17,37,4,21,154,8,21,154,8,19,63,8,19,63,8,17,37,30
19999 FORI=0TO37:READA:POKE49152+I,A:NEXT:REM**J/S**
30000 DATA 173,0,220,201,123,208,3,32,48,192,201,119,208,3,32,80
30010 DATA 195,201,107,208,3,32,48,192,201,103,208,3,32,80,195,32
30020 DATA 92,193,32,148,192,96
30030 FORI=0TO41:READA:POKE49200+I,A:NEXT:REM***ME LEFT***
30100 DATA 173,0,208,201,28,208,7,173,16,208,41,1,240,24,56,206
30110 DATA 0,208,206,0,208,206,0,208,206,0,208,208,9,173,16,208
30120 DATA 73,1,141,16,208,24,32,136,194,96
30130 FORI=0TO41:READA:POKE50000+I,A:NEXT:REM***ME RIGHT**
30200 DATA 173,0,208,201,48,208,7,173,16,208,41,1,208,24,24,238
30210 DATA 0,208,238,0,208,238,0,208,238,0,208,208,9,173,16,208
30220 DATA 73,1,141,16,208,24,32,236,194,96
50040 FORI=0TO23:READA:POKE49500+I,A:NEXT:REM***MOVE & CHANGE ALIEN*****
50070 DATA162,1,189,248,7,24,105,1,201
50080 DATA212,144,2,169,210,157,248,7
50090 DATA232,224,8,208,236,96,0
50100 FORI=0TO23:READA:POKE49800+I,A:NEXT:REM***CHANGE ME*****
50170 DATA162,0,189,248,7,24,105,1,201
50180 DATA225,144,2,169,217,157,248,7
50190 DATA232,224,1,208,236,96,0
50200 FORI=0TO23:READA:POKE49900+I,A:NEXT:REM***CHANGE ME*****
50270 DATA162,0,189,248,7,24,105,1,201
50280 DATA217,144,2,169,210,157,248,7
50290 DATA232,224,1,208,236,96,0
53000 FORI=0TO19:READA:POKE50100+I,A:NEXT:REM***ME  ↑**
53010 DATA173,1,208,201,60,240,12,206,1,208,206,1,208,206,1,208,206,1,208,96
53020 FORI=0TO19:READA:POKE50150+I,A:NEXT:REM***ME DOWN***
53030 DATA173,1,208,201,200,240,12,238,1,208,238,1,208,238,1,208,238,1,208,96
54999 FORI=0TO22:READA:POKE49300+I,A:NEXT:REM**JUMP SUB/ROU****
55000 DATA 32,24,196,32,86,196,32,135,196,32,179,196,96
55002 DATA32,16,197,32,69,197,32,128,197,96
55004 FORI=0TO222:READA:POKE50200+I,A:NEXT:REM**MOVE ALIEN**
55005 DATA 173,2,208,201,244,240,13,238,2,208,238,2,208,238,2,208
55010 DATA 238,2,208,96,169,55,141,149,192,169,196,141,150,192,96,173
55020 DATA 2,208,201,20,240,13,206,2,208,206,2,208,206,2,208,206
55030 DATA 2,208,96,169,24,141,149,192,169,196,141,150,192,96,173,4
55040 DATA 208,201,150,240,7,238,4,208,238,4,208,96,169,111,141,152
55050 DATA 192,169,196,141,153,192,96,173,4,208,201,24,240,7,206,4
55060 DATA 208,206,4,208,96,169,86,141,152,192,169,196,141,153,192,173
55070 DATA 6,208,201,60,240,4,238,6,208,96,169,157,141,155,192,169
55080 DATA196,141,156,192,96,173,6,208,201,1,240,4,206,6,208,96
55090 DATA169,135,141,155,192,169,196,141,156,192,96,173,8,208,201,231
55100 DATA240,16,238,8,208,238,8,208,238,8,208,234,234,234,234,234
55110 DATA234,96,169,213,141,158,192,169,196,141,159,192,96,173,8,208
55120 DATA201,72,240,16,206,8,208,206,8,208,206,8,208,234,234,234
55130 DATA234,234,234,96,169,179,141,158,192,169,196,141,159,192,96
55131 FORI=0TO167:READA:POKE50423+I,A:NEXT:REM***ALIENS SC 2***
55135 DATA173,13,208,201,50,240,7,206,13,208,206,13,208,96,169,16
55140 DATA141,162,192,169,197,141,163,192,96,173,13,208,201,190,240,7
```

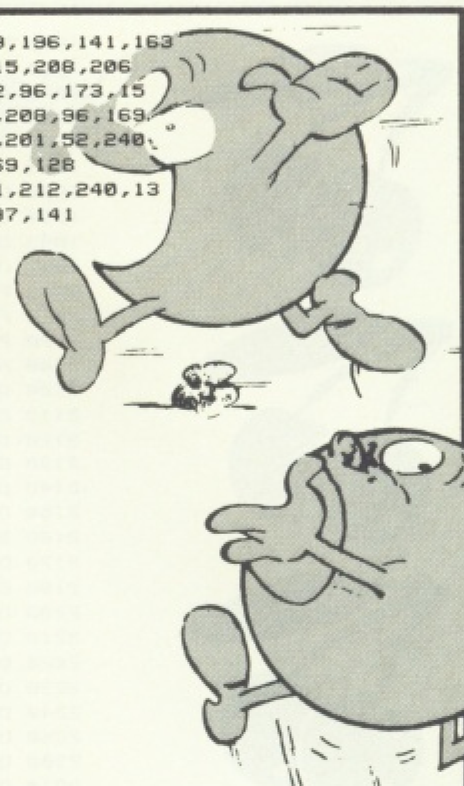




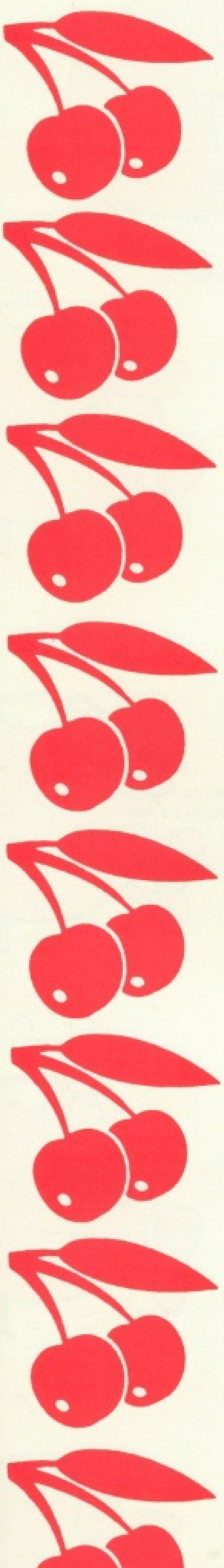
```
55170 DATA 238,13,208,238,13,208,96,169,247,141,162,192,169,196,141,163
55180 DATA 192,96,173,15,208,201,50,240,10,206,15,208,206,15,208,206
55190 DATA 15,208,96,169,69,141,165,192,169,197,141,166,192,96,173,15
55200 DATA 208,201,215,240,10,238,15,208,238,15,208,238,15,208,96,169
55210 DATA 41,141,165,192,169,197,141,166,192,96,173,7,208,201,52,240
55220 DATA 13,206,7,208,206,7,208,206,7,208,206,7,208,96,169,128
55230 DATA 141,168,192,169,197,141,169,192,96,173,7,208,201,212,240,13
55240 DATA 238,7,208,238,7,208,238,7,208,238,7,208,96,169,97,141
55250 DATA 168,192,169,197,141,169,192,96
59999 REM CLR
60000 PRINT"■":POKE53280,0:POKE53281,0:POKE53272,29
60009 REM RED-CRD BLU YEL
60010 PRINT"■■     CHERRY PICKER■...BY ■FRANK TOUT.   "
60011 REM CYN
60012 PRINT"■              INSTRUCTIONS              ";
60019 REM   2*CRD-PUR
60020 PRINT"■■■  YOUR TASK IS TO GUIDE FREDDIE THROUGH ";
60024 REM GRN
60025 PRINT"■  5 WAVES,COLLECTING CHERRYS S YOU GO,   ";
60029 REM BLU
60030 PRINT"■  BUT BEWARE OF ALL THE SPIDERS ETC..    ";
60039 REM YEL
60040 PRINT"■  THAT ARE THERE TO STOP YOU.            "
60059 REM ORN
60060 PRINT"■   PLUG YOUR JOYSTICK INTO PORT TWO      ";
60069 REM BWN
60070 PRINT"■   LEFT = LEFT :RIGHT = RIGHT :F/B TO    ";
60071 REM LRD
60072 PRINT"■   JUMP.YOU CAN ALSO GUIDE FREDDIE AFTER";
60073 REM GR1
60074 PRINT"■   HE HAS JUMPED,SO IF YOU WANT TO JUMP  ";
60075 REM GR2
60076 PRINT"■   A LONG WAY,KEEP THE JOYSTICK OVER TO  ";
60077 REM LGN
60078 PRINT"■   THE LEFT OR RIGHT,FOR A SHORT JUMP    ";
60079 REM LBL
60080 PRINT"■   RELEASE THE JOYSTICK,AND TO JUMP UP   ";
60082 REM GR3
60083 PRINT"■   DO NOT MOVE THE JOYSTICK AT ALL.      ":POKE198,0
60084 REM GRN BLU YEL
60085 PRINT"■       THERES MORE ■PRESS ■SPACE"
61000 GETA$:IFA$=""THEN61000
61010 IFA$=" "THEN61110
61050 GOTO61000
61109 REM CLR-  2*CRD-WHT
61110 PRINT"■■■■  YOU SCORE 100 PTS FOR EACH CHERRY    ";
61119 REM RED
61120 PRINT"■  YOU COLLECT AND YOU HAVE 3 LIVES WITH  ";
61129 REM PUR
61130 PRINT"■  AN EXTRA LIFE ON WAVES 3 AND 5.      ":POKE198,0
61139 REM   3*CRD-BLU YEL RED PUR
61140 PRINT"■■■■ NOW PRESS ■SPACE ■TO LOAD ■CHERRY PICKER "
61150 GETA$:IFA$=""THEN61150
61160 IFA$=" "THENPOKE198,2:POKE631,13:LOAD
61170 GOTO61150
READY.
```

Program Listing Part 3

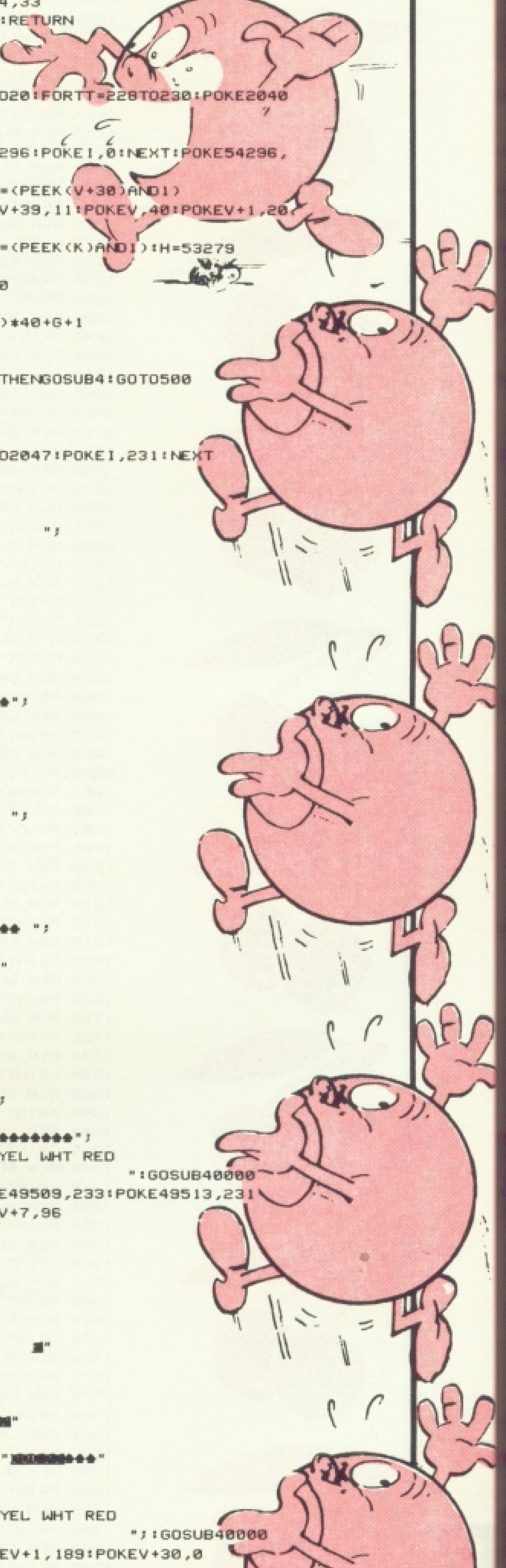
```
0 X=0:POKE53272,29:SYS36864:POKE54296,15:GOSUB60000:CLR:LI=3:SC=0:GOTO100
1 IFFNB(B)=65THENRETURN
2 SYSM2:GOSUB26:RETURN
4 G=782:X=X+1:SYSM3:SYSM1:GOSUB24:GOSUB10:IFX>7THENX=0:G=862:RETURN
6 GOTO4
10 IFFNC(C)=9THENGOSUB15:RETURN
11 J=FND(D):IFFNA(B)=1THENGOSUB21:GOSUB410:RETURN
12 IFFNB(B)=RTHENPOKEJ,W:SC=SC+Q:CH=CH+1:GOSUB32:GOSUB20:GOSUB40010:RETURN
13 IFFNB(B)>67THENGOSUB21:GOSUB410:RETURN
14 RETURN
15 J=FNE(E):RETURN
18 RETURN
20 POKES1+14,32:POKES2+14,44:POKES3+14,240:POKES1+14,33:RETURN
20 REM CLR- 12*CRD RED
21 GOSUB60:PRINT"■■■■■■■■■■■■■"TAB(14)"■WHOOPS":LI=LI-1
22 POKES1+7,32:POKES2+7,43:POKES3+7,3:POKES1+7,33
23 POKES1+14,32:POKES2+14,43:POKES3+14,4:POKES1+14,33:FORT=0TO2000:NEXT:RETURN
24 POKES1+7,16:POKES2+7,20:POKES3+7,255-PEEK(U):POKES1+7,17:RETURN
26 POKES1+7,16:POKES2+7,20:POKES3+7,255-PEEK(U):POKES1+7,17:RETURN
32 IFCH>8THENLE=LE+1:GOSUB410
33 RETURN
```

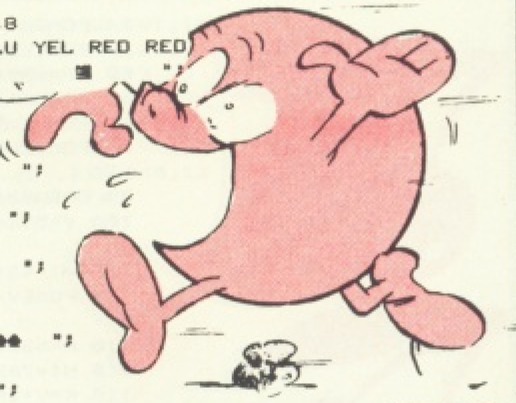






```
34 POKES1+14,32:POKES2+14,44:POKES3+14,TT-SO:POKES1+14,33
36 POKES1+7,32:POKES2+7,15:POKES3+7,TT-SO:POKES1+7,33:RETURN
60 SO=200:FORYY=PEEK(V+1)TO190STEP10:FORTT=225TO227
61 POKEV+1,YY:GOSUB34
61 REM CLR
62 POKE2040,TT:NEXTTT,YY:PRINT"■":POKEV+21,1: FORT=0TO20:FORTT=228TO230:POKE2040
,TT
64 GOSUB34:SO=100:NEXTTT,T:RETURN
100 V=53248:S1=54276:S2=54277:S3=54273:FORI=54272TO54296:POKEI,0:NEXT:POKE54296,
15
102 M1=49152:M2=50150:L=56320:M3=50100:LE=1:DEFFNA(A)=(PEEK(V+30)AND1)
104 POKEV+21,1:POKEV+28,1:POKEV+37,10:POKEV+38,6:POKEV+39,11:POKEV,40:POKEV+1,20
0
110 H=53279:K=53278:DEFFNA(A)=(PEEK(H)AND1):DEFFNA(B)=(PEEK(K)AND1):H=53279
115 H1=782:G=862:W=32:Q=100:U=53249:R=67:K=V+30:O=112
116 K=V+30:DEFFNB(B)=PEEK(J):DEFFNC(C)=PEEK(V+16):SC=0
130 DEFFND(D)=INT(PEEK(V)/8)+INT(PEEK(U)/8)*40+G
135 DEFFNE(E)=INT(PEEK(V)/8)+INT(255)/8+INT(PEEK(U)/8)*40+G+1
400 POKEV+1,190:LE=1
410 ONLEGOSUB900,1100,1200,1300,1400,900
500 SYSM1:GOSUB10:GOSUB1:P=PEEK(L):IFP<OANDPEEK(J)=65THENGOSUB4:GOTO500
525 IFLI<1THENGOSUB49999:GOTO0
550 GOTO500
899 REM CLR-RED-   2*CRD
900 PRINT"■■■";:POKEV,40:POKEV+1,191:CH=0:FORI=2041TO2047:POKEI,231:NEXT
901 POKE2040,210:POKEV+1,186:POKE49312,96:G=862
1016 PRINT"♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠"
1017 REM LRD GR3 LRD GRN LRD YEL
1018 PRINT" ■—      ■◣  ■—         ■|       ■—    ■◣              ";
1019 REM GRN
1020 PRINT"■                        |"
1021 REM RED GRN
1022 PRINT"■♠♠♠ ■                        |"
1024 PRINT"                      |"
1025 REM BLU GRN BLU
1026 PRINT"■           ||          ■|■          ||"
1026 REM RED GRN RED
1027 PRINT"■♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠■|     ■♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠";
1027 REM YEL LRD YEL LRD GRN LRD GRN LRD BLU
1028 PRINT"  ■◣    ■—  ■◣      ■—  ■|     ■—    ■| ■— ■◣"
1029 REM GRN
1030 PRINT"■                  |         |"
1031 REM RED GRN RED
1032 PRINT"■                  ♠♠♠■|          |■♠♠♠          ";
1033 REM LBL GRN
1034 PRINT"                      ■—            ■|"
1035 REM LBL BLU
1036 PRINT"■                                  —   ■|||||||"
1037 REM RED WHT RED WHT RED
1038 PRINT"■♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠■■♠♠♠      ♠♠♠♠♠♠■■♠♠♠♠♠♠ ";
1039 REM LRD BLU LRD GRN LRD BLU LGN
1040 PRINT"          ■—  ■◣ ■—           ■|        ■  ■◣    ■||"
1042 PRINT"                         |                  |"
1043 REM RED GRN RED GRN
1044 PRINT"■♠♠♠                      ■|        ■♠♠♠     ■|"
1045 REM LBL
1046 PRINT"■                            —              —"
1047 REM BLU
1048 PRINT"■                                         ||||||  ";
1049 REM RED BLU RED WHT RED BLU RED WHT RED
1050 PRINT"■♠♠♠♠♠♠♠♠♠♠♠♠♠■||■♠♠♠♠♠■■♠♠♠♠♠■||■♠♠■■♠♠♠♠♠♠♠♠♠♠";
1050 REM CYN PUR GRN BLU YEL WHT RED CYN PUR GRN BLU YEL WHT RED
1051 PRINT"               ◣THE ■OP■■OLE■RS ■LOA■ER                ";:GOSUB40000
1055 POKEV+28,1:POKEV+21,31:POKEV+2,28:POKEV+3,50:POKE49509,233:POKE49513,231
1057 POKEV+4,50:POKEV+5,146:POKEV+16,8:POKEV+6,1:POKEV+7,96
1059 POKEV+8,108:POKEV+9,196:POKEV+30,0:RETURN
1099 REM CLR-   2*CRD    4*CRD
1100 PRINT"■■■";:B$="■■■■":CH=0:LE=2:G=862
1109 REM ORN
1110 A$="■♠♠♠♠    ♠♠♠♠♠    ♠♠♠    ♠♠♠    ♠♠♠♠♠♠♠♠♠♠♠♠♠"
1111 REM RED WHT RED WHT RED WHT RED WHT
1112 C$="■          — ■     ■—■     ■—■     ■                  ■"
1112 REM CRU-BLU-   4*CRR    5*CRR    3*CRR    3*CRR
1113 D$="■■■■■||■■■■■||■■■■||■■■■"
1114 REM HOM-   4*CRD
1115 PRINTA$;C$;B$;A$;C$;B$;A$;D$:PRINT"■■■■"
1119 REM ORN    5*CRD    5*CRD
1120 PRINTTAB(4)"■♠♠♠":PRINTTAB(31)"■■■■♠♠♠♠"TAB(44)"■■■■♠♠♠♠"
1149 REM HOM- 21*CRD
1150 PRINT"■■■■■■■■■■■■■■■■■■■■■";
1150 REM CYN PUR GRN BLU YEL WHT RED CYN PUR GRN BLU YEL WHT RED
1151 PRINT"                ◣THE ■GH■DS◣THE ■R■D■M              ";:GOSUB40000
1155 POKEV+8,108:POKEV+9,192:POKEV+21,17:POKEV,28:POKEV+1,189:POKEV+30,0
1160 POKE2040,210:POKEV+21,223:POKEV+16,8
1165 POKEV+12,120:POKEV+13,100:POKEV+14,215:POKEV+15,150:POKEV+6,1:POKEV+7,100
1170 POKEV+2,200:POKEV+3,144:POKEV+4,150:POKEV+5,95:POKE49312,234
1172 POKE49509,236:POKE49513,233:RETURN
1199 REM CLR
```

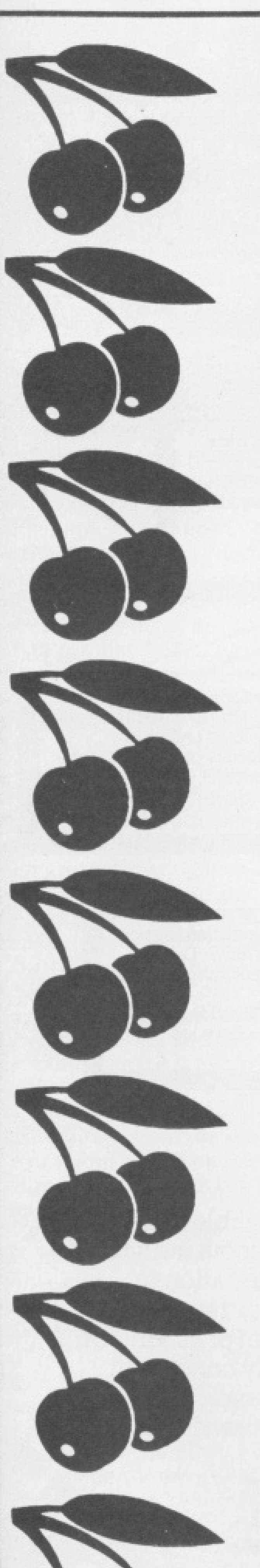




```
1200 PRINT"";:CH=0:G=862:POKEV,28:POKEV+1,53:POKEV+16,8
1209 REM BWN RED GRN CYN PUR GRN BLU YEL RED PUR GRN BLU YEL RED RED
1210 PRINT"          -   THE  CLOCK  SHOP        ";
1211 REM BWN GRN
1212 PRINT"              ||||||||||||            ";
1213 REM BWN LBL GR1 GRN GR1
1214 PRINT"                                      ";
1215 REM BWN RED GRN BWN
1216 PRINT"                                      ";
1217 REM RED BWN RED BWN GRN
1218 PRINT"                        |             ";
1220 PRINT"                   |       |          ";
1224 REM BWN RED BWN GRN BWN GRN BWN
1225 PRINT"                                      ";
1229 REM GRN RED GRN
1230 PRINT"                                      ";
1235 PRINT"                   |       |          ";
1239 REM BWN GRN BWN GRN
1240 PRINT"                                      ";
1244 REM BWN RED GRN BWN
1245 PRINT"                             |        ";
1249 REM GRN
1250 PRINT"                   |       |    |     ";
1254 REM BWN RED BWN GRN BWN GRN
1255 PRINT"                                 |    ";
1259 REM BWN GRN GR1 RED GRN LBL
1260 PRINT"                                      ";
1264 REM GRN RED GRN BWN RED
1265 PRINT"                                      ";
1269 REM BWN CYN GRN BWN GRN
1270 PRINT"                                      ";
1274 REM BWN GRN
1275 PRINT"                           |          ";
1277 PRINT"                   |       |          ";
1279 REM BWN RED BWN LBL BWN GRN BWN
1280 PRINT"                                      ";
1284 REM GRN
1285 PRINT"                           |          ";
1286 REM LBL
1287 PRINT"                           -          ";
1289 REM BWN BLU BWN BLU BWN BLU BWN
1290 PRINT"        |||||||||           ||        ";
1291 GOSUB40000:POKEV+21,255:POKEV+2,244:POKEV+3,102:POKEV+4,88:POKEV+5,79
1292 POKEV+8,225:POKEV+9,199:POKEV+10,238:POKEV+11,149:POKEV+12,130:POKEV+13,64
1293 POKEV+14,70:POKEV+15,200:POKE2040,211
1295 POKE49509,243:POKE49513,237:POKEV+30,0:RETURN
1299 REM CLR-RED   2*CRD
1300 PRINT" -    -     -    --    -     -        ";:G=862:CH=0
1314 REM BLU
1315 PRINT"                                      ";
1319 REM CRD-WHT BLU
1320 PRINT"                                      ";
1324 REM WHT GRN
1325 PRINT"                           |          ";
1329 REM BWN BLU GRN
1330 PRINT"                                      ";
1334 REM BLU BWN PUR
1335 PRINT"                                      ";
1339 REM BLU BWN PUR
1340 PRINT"                           |          ";
1344 REM BWN PUR LBL BLU
1345 PRINT"               |                      ";
1349 REM PUR BLU BWN
1350 PRINT"               |                      ";
1354 REM BLU PUR BWN
1355 PRINT"                                      ";
1359 REM BLU PUR LBL
1360 PRINT"                                      ";
1364 REM BWN PUR LBL
1365 PRINT"  -                                   ";
1369 REM PUR WHT BWN
1370 PRINT"                                      ";
1374 REM BLU PUR WHT BLU
1375 PRINT"                                      ";
1379 REM LBL
1380 PRINT"                                      ";
1384 REM CRD
1385 PRINT"";
1385 REM BLU LBL BLU
1386 PRINT"        ||||||||||||||||||||||||      ";
1388 REM LBL
1389 PRINT"                                      ";
1389 REM CYN PUR GRN RED GRN BLU YEL RED CYN PUR GRN BLU RED
1390 PRINT"       THE  CLOCK  SHOP  2              ":GOSUB40000
1391 POKEV+21,255:POKEV,28:POKEV+1,186:POKE2040,210:POKEV+16,8
1392 POKEV+2,252:POKEV+3,90:POKEV+4,130:POKEV+5,130:POKEV+6,10:POKEV+7,70
1395 POKEV+8,78:POKEV+9,50:POKE49509,243:POKE49513,237:POKEV+30,0:RETURN
1399 REM CLR-  2*CRD
```



```
1400 PRINT"■■■";:G=862:CH=0
1404 REM PUR BLU RED   2*CRD-  2*CRL-PUR
1405 R$="■◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆■||||||||||":S$="■■■■■◆◆◆"
1406 REM PUR
1407 T$="■          ◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆"
1408 REM GRN CRD-CRL CRD-CRL CRD-CRL CRD-CRL CRD-CRL CRD-CRL CRD-CRL CRD-CRL
1409 REM CRD-CRL CRD-CRL CRD-CRL CRD-CRL CRD-CRL-LBL
1410 U$="■■■■■■■■■■■■■■■■■■■■■■■■■■■■"
1450 PRINTT$:PRINTTAB(19)S$:PRINTTAB(19)S$:PRINTTAB(19)S$:PRINTTAB(19)S$
1454 REM   2*CRD HOM-  5*CRD
1455 PRINTTAB(19)S$:PRINT"■■"R$:PRINT"■■■■■■":PRINTTAB(10)S$:PRINTTAB(8)S$
1456 REM HOM-  2*CRD
1457 PRINTTAB(12)S$:PRINTTAB(5)S$:PRINT"■■■":PRINTTAB(15)U$
1464 REM HOM-  2*CRD
1465 PRINT"■■■":PRINTTAB(26)U$:POKEV+21,255:POKEV,28:POKEV+1,190
1470 POKEV+2,252:POKEV+3,50:POKEV+4,70:POKEV+5,80:POKEV+6,0:POKEV+7,100
1475 POKEV+16,8:POKEV+8,90:POKEV+9,150:POKEV+10,200:POKEV+11,111
1485 POKEV+12,60:POKEV+13,170:POKEV+14,100:POKEV+15,182
1490 FORI=2041TO2047:POKEI,243:NEXT
1496 POKE49509,247:POKE49513,243:POKE49312,234:POKEV+30,0:
1497 REM   5*CRD
1498 PRINT"■■■■■":GOSUB40000:RETURN
39999 REM GR2 LGN LBL GR3
40000 PRINTTAB(3)"■SC"TAB(15)"■LI"TAB(20)"■LE"TAB(26)"■HI"
40009 REM HOM- 22*CRD
40010 PRINT"■■■■■■■■■■■■■■■■■■■■■■■":IFSC>HITHENHI=SC
40011 PRINTTAB(6);SC;TAB(17);LI;TAB(22);LE;TAB(28);HI
41111 RETURN
49998 REM CLR
49999 PRINT"■":POKEV+21,0
50010 G$="                 GAME  OVER"
50014 REM HOM-  6*CRD
50015 FORT=0TO14:PRINT"■■■■■■■";:FORTT=0TO9:PRINTG$:POKE646,INT(RND(1)*15)
50020 GOSUB50030:NEXTTT,T:GOSUB61000:RETURN
50030 POKES1+7,32:POKES2+7,13:POKES3+7,TT:POKES1+7,33:RETURN
59999 REM CLR-CRD-RED
60000 POKE53280,0:POKE53281,0:PRINT"■■■";
60014 REM BLU RED
60015 PRINT"       ■||| |  | |||| |||  |||  |   | ■◆     ";
60019 REM BLU RED BLU RED BLU RED
60020 PRINT"    ■|■◆◆◆ ◆  ◆ ◆◆◆◆ ◆◆◆■|■ ◆◆◆■|■ ◆    ◆ ◆     ";
60025 PRINT"    ◆     ◆ ◆◆     ◆ ◆ ◆ ◆◆   ◆◆    ";
60029 REM BLU RED BLU RED BLU RED BLU RED BLU RED
60030 PRINT"    ◆     ◆■|■◆ ◆■||  ■◆■|■◆ ◆■|■◆ ◆■||■◆ ◆     ";
60035 PRINT"    ◆     ◆◆◆◆ ◆◆◆  ◆◆◆  ◆◆◆   ◆◆◆  ◆    ";
60039 REM GRN RED GRN RED BLU RED BLU GRN RED GRN RED
60040 PRINT"    ◆     ◆■ˎˎ■◆ ◆■ˎˎ  ■◆ ◆■|■◆ ◆■|  ■ˎ■◆■ˎ  ■◆     ";
60044 REM BLU RED BLU RED
60045 PRINT"    ◆■|||■◆  ◆ ◆■|||■◆ ◆◆ ◆ ◆◆   ◆   ◆    ";
60049 REM GRN RED
60050 PRINT"    ■ˎ■◆◆◆ ◆  ◆ ◆◆◆◆ ◆  ◆  ◆   ◆    ◆     ";
60054 REM GRN RED
60055 PRINT"    ■ˎˎˎ ˎ  ˎ ˎˎˎˎ ˎ  ˎ  ˎ    ■◆   ◆     ";
60059 REM BLU RED
60060 PRINT"   ■||||||||||||||||||||||||||||■◆   ◆     ";
60065 PRINT"   ◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆   ◆    ";
60069 REM BLU RED
60070 PRINT" ◆ ■|||  |||   ||| |  | |||| |||   ■◆     ";
60074 REM BLU RED BLU RED BLU RED
60075 PRINT" ◆ ◆◆◆■| ■◆◆◆  ■|■◆◆◆ ◆  ◆ ◆◆◆◆ ◆◆◆■|■  ◆     ";
60079 REM BLU RED
60080 PRINT" ◆   ◆  ◆  ◆   ◆   ◆ ■|■◆ ◆    ◆  ◆  ◆     ";
60084 REM BLU RED BLU RED BLU RED BLU RED
60085 PRINT" ◆ ◆■||■◆  ◆   ◆    ◆■|■◆ ◆■||  ■◆■||■◆  ◆     ";
60089 REM BLU RED
60090 PRINT" ◆ ◆◆◆   ◆   ◆    ◆◆■|■  ◆◆◆  ◆◆◆  ◆     ";
60094 REM BLU RED BLU RED
60095 PRINT" ◆ ◆     ◆   ◆    ◆ ◆■|■ ◆    ◆ ◆■|■  ◆     ";
60099 REM BLU RED BLU RED BLU RED BLU RED BLU RED
60100 PRINT" ◆ ◆     ■|■◆■|■  ◆■|||  ■◆  ◆ ◆■|||  ■◆ ◆◆■|■◆     ";
60105 PRINT" ◆ ◆     ◆◆◆   ◆◆◆ ◆  ◆ ◆◆◆◆ ◆  ◆◆◆◆     ";
60109 REM GRN
60110 PRINT" ◆ ■ˎ     ˎˎˎ   ˎˎˎ ˎ  ˎ ˎˎˎˎ ˎ  ˎˎˎˎ     ";
60110 REM RED BLU
60111 PRINT"   ■◆■|||||||||||||||||||||||||||||||||| ";:POKE198,0
60114 REM RED
60115 PRINT"■  ◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆"
60124 REM HOM PUR BLU YEL CYN GRN BLU WHT PUR YEL BLU
60125 PRINT"■ ";X$;",S  ■H■I■G■H ■S■C■O■R■E■";HI
60200 GETQ$:IFQ$=""THEN60200
60250 IFQ$=" "THENRETURN
60299 GOTO60200
61000 IFSC<HITHENRETURN
61004 REM CLR-  5*CRD-RED
61005 PRINT"■■■■■■      PLEASE STATE YOUR NAME"
61009 REM BLU YEL
61010 INPUT"■>■";X$
61020 RETURN
READY.
```





VIZASTAR 64

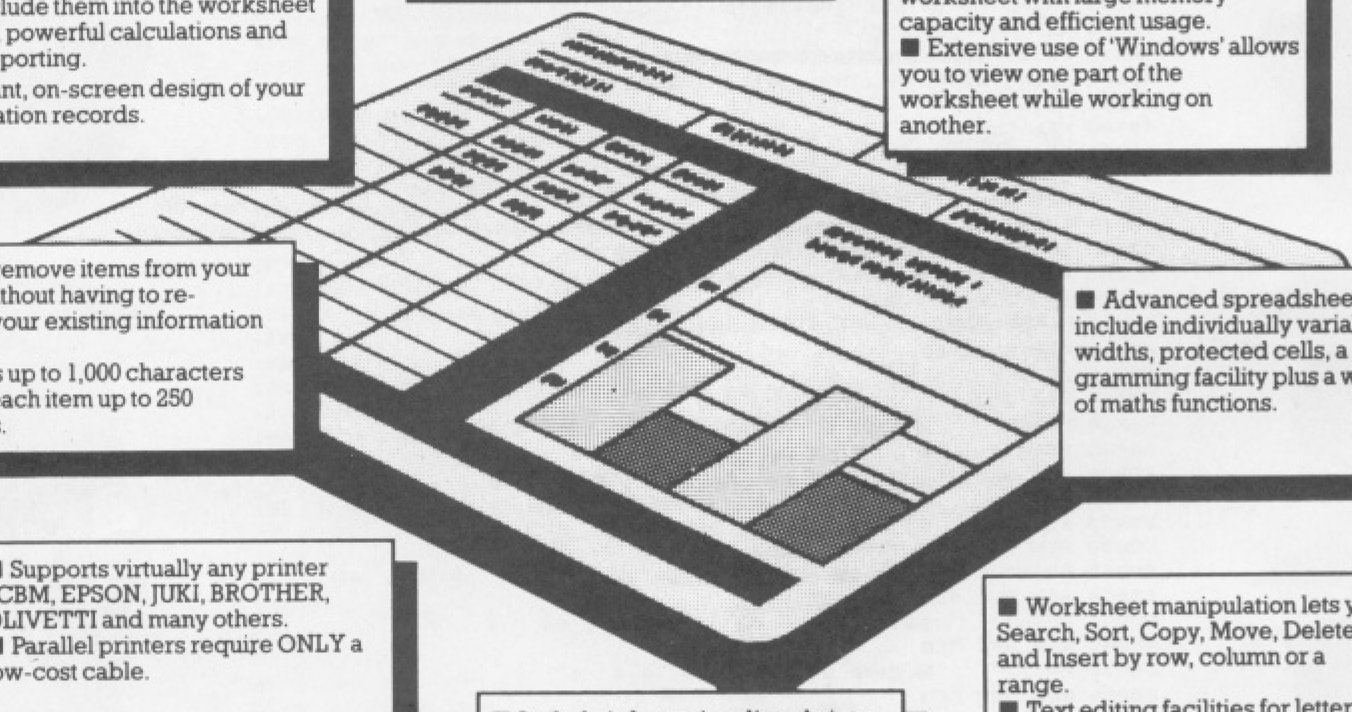
THE INFORMATION PROCESSOR

■ Spreadsheet ■ Database ■ Graphics

■ One word menu lines for easy selection of facilities.
■ The high resolution grid pattern frames the worksheet giving an easy-to read display.

■ Extract from your information files and include them into the worksheet for fast, powerful calculations and easy reporting.
■ Instant, on-screen design of your information records.

■ A 1000 Row by 64 Column wide worksheet with large memory capacity and efficient usage.
■ Extensive use of 'Windows' allows you to view one part of the worksheet while working on another.



■ Add or remove items from your records without having to re-construct your existing information records.
■ Records up to 1,000 characters long with each item up to 250 characters.

■ Advanced spreadsheet features include individually variable column widths, protected cells, a programming facility plus a wide range of maths functions.

■ Supports virtually any printer –CBM, EPSON, JUKI, BROTHER, OLIVETTI and many others.
■ Parallel printers require ONLY a low-cost cable.

■ Worksheet manipulation lets you Search, Sort, Copy, Move, Delete and Insert by row, column or a range.
■ Text editing facilities for letters, invoices, report headings etc.

■ Include information directly into the Worksheet from VizaWrite and almost any other system.
■ Simultaneous display of work-sheet and line or bar graphs.

VizaStar integrates three important electronic aids to your business or home office.
A fast and easy-to-use package – VizaStar is a comprehensive information processor that includes an electronic worksheet, information filing system and simultaneous on-screen graphics.

VizaStar has been designed to incorporate the very latest in user-friendliness – so now you can handle your all-important information in many new ways and in a quick and effective manner.

The information filing capabilities of VizaStar enable you to store your information onto disk in an ordered and instantly retrievable form. This means that you can quickly pick out individual or entire groups of related information to create lists, statements, labels or reports.

VizaStar is the ONLY program of its kind on the Commodore 64. Completely consistent with the VizaWrite word processor, VizaStar provides a totally comprehensive office system.

VizaStar costs just £99.95 (incl. VAT) from your computer dealer and selected stores. Please write for more details and the name and address of your nearest dealer.

AVAILABLE NOW- VIZASTAR 64
£99.95
(inc VAT)

VIZA SOFTWARE

VIZA SOFTWARE, 9 MANSION ROW,
BROMPTON, GILLINGHAM, KENT ME7 5SE
TEL: MEDWAY (0634) 813780
Dealer enquiries welcome.

GAMES MANSHIP

In part three of this series, David Rees shows you how to overcome the main drawback of BASIC games — lack of speed.

MOST BASIC GAMES LACK one vital commodity — speed. This means that the majority of BASIC arcade type games are slow and boring when implemented, and thus become unpopular. Machine code (including those routines given in the first two parts of this series) helps to speed things up, but BASIC running time is still the main limiting factor. To remedy this, part three concentrates on speed increases in this domain.

Some speed increases are obvious (e.g. removing spaces and putting several statements onto each line), but others need greater investigation. It is always a good idea to put the main routine (the section that is used constantly when the game is in progress) near to the start of the program. Not only does this mean that GOTO line numbers will be shorter, but exeucution time is saved. This is because each line in Commodore BASIC is found by tracing along from the start of the program, so the nearer it is to the start, the quicker it is found.

Style and logical layout are also important. If you put all eventualities in the main routine, the game will be very slow. It is best to strip the main routine to the bone, leaving only a PRINT, GET, GOTO and a few IF/THENs (with a few SYSs if machine code is to be used). To cope with the other needs of the game, simply GOTO or GOSUB separate routines. For instance, to test key presses, you only need a GET statement and an IF condition. This can decide if any key at all has been pressed, and if this is so, a separate routine can handle the combinations of presses (quick tip here: POKE 650,128 gives repeat for all keys, and POKE 650,0 gives normal repeat, while POKE 650,64 stops all repeating).

It is sensible to put all routines in order of frequency of use so that those needed most often can be accessed in less time. Thus, a key respond will be first in most cases, while a lose life routine would probably be last.

Usage of certain functions should be kept to a minimum. For example, the easy way to update the screen figures is to PRINT them with each routine cycle.

However, score and most certainly lives lost does not need this rapid update. It is more appropriate to print out each set of figures only when they change, thus saving time.

Listing

```
1 REM****************
2 REM*SCORE ROUTINE*
3 REM*     BY      *
4 REM* DAVID REES  *
5 REM****************
4000 FORN=0TO9:GETA$:NEXT
4009 REM*[CLR HOME]
4010 POKE56325,50:PRINT"■";:POKEV+21,0
4020 IFS(9)>=STHEN4200
4030 PRINT"YOU ARE IN THE TOP TEN SCORES"
4039 REM*[DOWN*2]
4040 PRINT"■■PLEASE ENTER YOUR NAME:"
4050 INPUTN$
4060 N$=LEFT$(N$,14)
4070 N=0
4080 FORN=0TO9:IFS>S(N)THEN4100
4090 NEXT
4100 FORM=9TONSTEP-1
4110 N$(M+1)=N$(M):S(M+1)=S(M):NEXT
4120 S(N)=S:N$(N)=N$
4199 REM*[CLR HOME]
4200 PRINT"■";
4209 REM*[RVS ON][L.BLUE]
4210 PRINTSPC(15)"■■ASTEROIDS"
4219 REM*[PURPLE][DOWN]
4220 PRINTSPC(16)"■SCORES■"
4229 REM*[WHITE]
4230 PRINT" ■1 "N$(0)
4239 REM*[UP][DOWN][GREEN]
4240 PRINT"■"TAB(20);S(0)"■■"
4250 FORN=1TO9
4260 PRINTN+1;N$(N)
4269 REM*[UP][DOWN]
4270 PRINT"■"TAB(20);S(N)"■"
4280 NEXT
4289 REM*[RVS ON][L.BLUE]
4290 PRINT"■■PRESS ANY KEY TO CONTINUE"
4300 GETA$:IFA$=""THEN4300
4399 REM*[CLR HOME]
```

Instructions

Hopefully, your own ingenuity and the information provided in these articles has lead you to create a fast, interesting game. However, many people falter at the next, final step. A game is only fun if it is easy to use and has some extra incentives attached, but many programmers miss this point in the rush for their family's and friends' acclaim.

One of the most important extras is the instructions. Essentials, such as which key to press, should be included but a friendly, well set out introduction adds polish to the game. Options at the end of the game should not be ignored. If somebody wants to have another game, he should be allowed to do so easily, by following clearly set out instructions, rather than by RUNning the program again.

Finally, a player always likes to seem important, no matter how well he or she did. The best way to accomplish this is to use a score table. Listing 1 gives a score program for the Commodore, and allowances have been made so it can be easily integrated into a games program. Colour produced by the program is not shown in the screen dump, but it is still a powerful tool. Colour can be used for highlighting, and can easily make the game title and top score outstanding.

Having produced what is in your view *the* game of 1985 how do you convert your hobby into an arguably lucrative profession? In search of an answer, Alison Hjul spoke to some of the leading software houses.

SO YOU'VE WRITTEN A GAME

IT'S CERTAINLY TOUGH AT THE TOP. All the software houses I spoke to will look at any game submitted to them, but the standard is very high. "Everything sent to us gets looked at", insists Jeremy Cooke of Virgin Games. But, "The new market requires a high standard. It's increasingly difficult to find good stuff." Virgin accept approximately 5% of the games submitted to them; this is about average. A'n'F, for example, receive about 150 to 200 games a year of which around 4% are accepted. Other are less generous: Anirog accept about one game a year. How, then, do you qualify?

First steps

The software house moguls disagree on the form in which they wish to receive your game initially. Jeffrey Heath of Activision believes that the idea behind the game rather than its actual content is paramount. Activision would then pass it on to their European designers and, if the idea is approved, the programmer would be invited to discuss it further. Other companies, such as Quicksilva, would prefer to see the completed game.

However, most like to see a demonstration version. Roger Gamon of Anirog thinks it is necessary to see some concrete evidence of the game so his software experts can 'see all potential aspects of the game'.

Selection procedures

All the software houses I spoke to have different selection procedures. They vary from one resident software expert employed solely for the purpose of assessing submitted games to a team of reviewers. To quote Sandy Marchant of Bubble Bus, "A review body of 4 or 5 decides if the idea needs to be pursued".

If your game is given the 'thumbs down', it will be returned (but there's always another plan of attack - read



on). And don't raise your hopes too high even if you are given the initial stamp of approval. Mike Fitzgerald of A'n'F says that 30 or 40 of the games produced by them are never finalised. However, if the contract is withdrawn, they return the copyright to the programmer.

Very few games are published in their initial state. "We would accept very few games as presented", says Mike Fitzgerald. All software companies have teams of experts who enhance graphics and sound, for example. Mike Fitzgerald reckons that 4 to 6 weeks usually pass from the time of receiving a game to the time it's marketed.

What's in a game

Originality scores top marks, Jeremy Cooke says, "To some extent, there are still too many people sending in a straight rip off. It's like the music business where people copy a Paul McCartney song and then wonder why they're not successful." Like all software houses, Virgin are very anxious to hear from programmers with novel ideas.

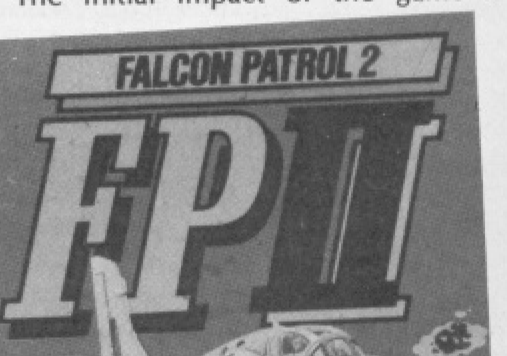
But a few borrowed routines may be acceptable — as long as this isn't carried to excess. Mike Fitzgerald told me: "If somebody actually disembowelled Jet Set Willy, I'd turn it down. But, for example, in Krazy Kong, there is a routine with things rolling down a girder. Now, if somebody used the same routine in a game submitted, I wouldn't turn it down because of this."

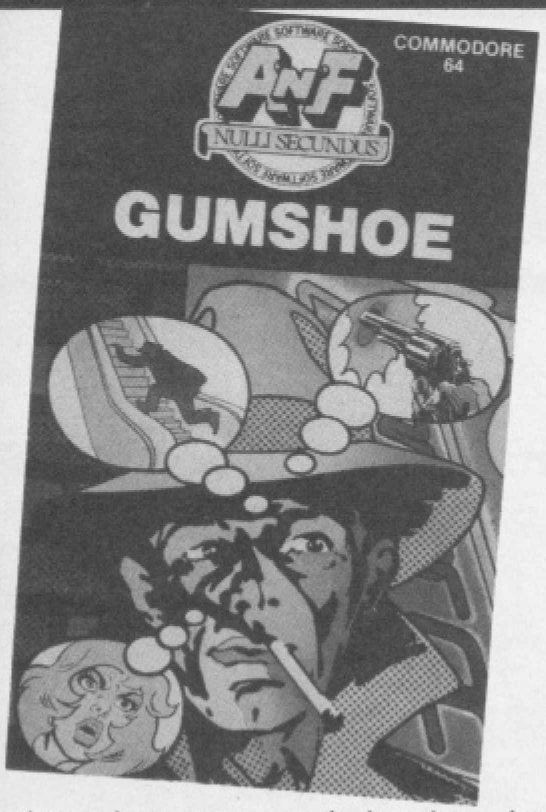
Original ideas are hard to come by. "We haven't had any really good original ideas sent in." said Roger Gamon. So, what else helps sell your game to the 'powers-that-be'?

Addiction is also very important. It is described by Jeremy Cooke as, "That magic quality whereby it's easy to start a game but difficult to keep going" or, simply, as 'playability', by Sandy Marchant of Bubble Bus.

Games written in BASIC are generally unpopular. Roger Gamon believes that some games (although strategy games rather than arcade games) 'can be written quite well in BASIC' but others believe, that games written in BASIC are given a bad press. "The punters tear it apart — they don't like it", says Jeremy Cooke. Mark Eyles of Quicksilva was the most outspoken in his condemnation of BASIC games: "I can't think of any program written in BASIC good enough to publish; it would need to be in machine code".

The initial impact of the game is





obviously important and, therefore, the sound and graphics should be of a high standard. But, as Mark Eyles pointed out, anybody submitting a game to a software house is assumed to be of a high technical standard — and software houses employ teams of people to enhance sound and graphics.

But, technical ability and creativity don't always go hand in hand — "Often good programmers aren't always the people with good ideas", says Jeremy Cooke. An exceptional programmer will be welcomed and encouraged as would an excellent game. "If the game is technically very good — say, an amazing version of hangman — although we're not interested in the game, we might use the programmer for conversion work or feed him with ideas".

Money, money, money...

So, your game — or your potential as a programming genius — has been accepted. What are your rewards? It's a myth that, once you're regarded favourably by one of the top software houses, you're on a quick road to fame and fortune. "A programmer's earning capacity has been exagerrated", says Roger Gamon. Having written a successful program, a programmer can earn as much as £1-2,000 a month but success soon wanes: the life of a program is now very short — about 2-3 months. Some programmers can earn £10-20,000 a year whereas others will earn a mere one or two hundred for one game which has gained minimal success.

Most companies encourage their programmers to accept royalties but, in some circumstances, the programmer can receive a straight fee. Royalties present more of a risk. Mark Eyles says, "If the game does well, we both do well," — and, of course, vice versa! But royalties do offer higher potential earnings. Roger Gamon stated the example where one programmer, who would have received an outright fee of £1,500, earned £11,000 in commission.

Most companies are also quite prepared to offer their programmers advanced royalties — to assist in the purchase of equipment crucial to the development of their next masterpiece, for example.

Other assistance

It is not only to your financial advantage to be found and nurtured by one of the leading software houses. As Mark Eyles told me, "With the market as it is now, if a programmer is capable of producing a top selling game, we will keep with him. With the life of games being not as long as they once were, programmers need to keep churning out hits. A lot of programmers produce one amazing game. When we find a good programmer, we encourage them to do a follow-up straight away — to keep on working". Their service to their programmers includes regular contact through newsletters. They also provide them with equipment, as do most software houses. Their new programmers will also have a wealth of advice and technical expertise at hand.

The software houses are always on the look out for new blood. Anirog, for example, advertise for new programmers. "We are definitely on the look out for new programmers," says Roger Gamon. Their programming requirements exceed the availability. This said, it is very difficult to find programmers of the calibre required and many resort to other means. Says Jeffrey Heath, "Because we have such a superb supply of products from our parent company, the standards needs to be exceptionally high."

Points to remember:

- Originality —Software houses don't want another version of Pacman
- Presentation —Initial impact and documentation are very important.
- Addiction —A low boredom rating on a game means better value for money.
- Machine code —BASIC isn't strictly taboo but Machine Code is more efficient and professional.
- Sound & Graphics —Such details grab the software expert's attention.
- Technical ability —Maybe you haven't got an original idea in your head - but the software houses are always on the look out for technical expertise.

Final note

However, don't be disillusioned by the seemingly impossible odds you face should you decide to submit your game to one of the software houses. Just bear in mind a few important facts. First of all, choose your potential publisher with care. If your game is a fast, addictive arcade type game opt for a software company which trades in that sort of software, such as Anirog. If adventure is your forte, then Level 9 are likely to take more interest in it than, say Virgin Games. Secondly, remember that the software houses receive a very large input of games and so those with instant appeal are likely to grab their attention. Make sure your game is well presented and the documentation is clear and accurate. As we've learnt from experience, there's nothing worse than receiving a game with a sound and interesting idea when, due to minimal documentation, we can't decipher how the hell to play it!

Magazines

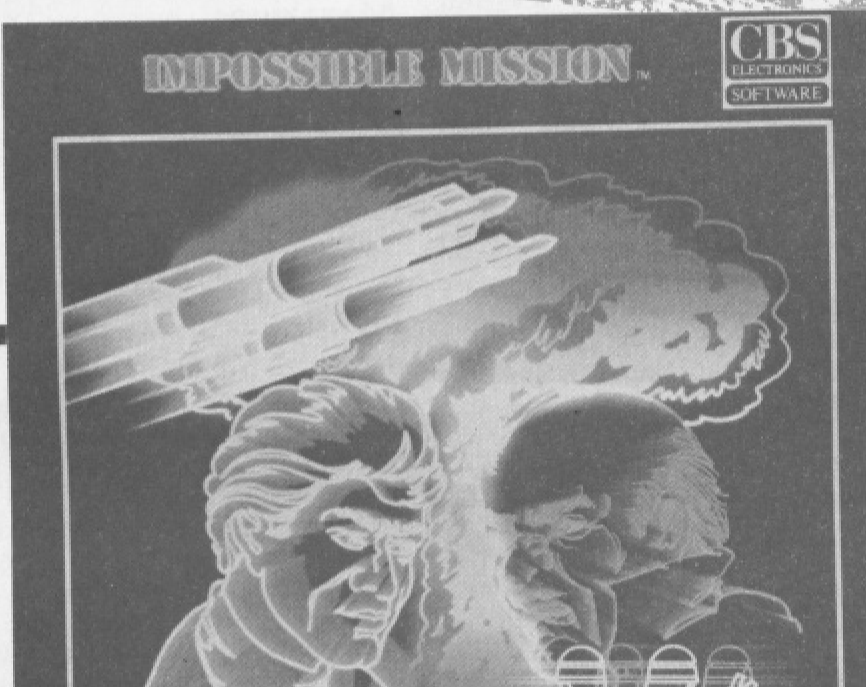
OK, so maybe the produce of your labour isn't a totally original game of Manic Miner standards. But it's good. Why should it only be reserved for the eyes of your nearest and dearest? Games players everywhere should bear witness to your semi-genius. All is not lost. Your deserved acclaim can yet exceed a look of admiration from little brother or a pat on the head from Aunt Alice. There's a wealth of computer magazines strewn across the shelves of your local newsagents (although only one of any note, I hasten to add!).

Magazines look for the same basic qualities in a submitted game as the software houses — originality, technical ability, good presentation and documentation (and a £10 note — Ed.). You should also take into consideration that we haven't a supply of technical experts on call to add amendments to your sound and graphics. So, it is probably more important that your game is totally bug free and does exactly what it says it does. The documentation should also be very clear and accurate as our readers who have to type in the game may not be as technically adept as you are — and we will be inundated with phone calls if they can't get the game to work!

DON'T BE SCARED. IT'S N

MESSAGE READS

- ALERT: AGENT 4125. *INFORMATION FROM AGENCY COMPUTER.*
- AFFIRMATIVE: *WORLD UNDER THREAT OF NUCLEAR ANNIHILATION.*
- MISSION: *URGENT FOIL PLOT.*



Out now on Epyx
from CBS Computer Software.

IMPOSSIBLE MISSION
PIT STOP
BREAKDANCE
TEMPLE OF APSHAI
SILICON WARRIOR
JUMPMAN
DRAGONRIDERS OF PERN
LUNAR OUTPOST

They make everything else
look just plain soft.

PROGRAMMING PROJECTS

Why tear your hair out for hours over some newspaper's infernal crossword when you can create your own with the aid of your 64? Garry Marshall shows you how.

A CROSSWORD IS A NATural arena for developing and using techniques for handling strings and arrays. The computer can be made to display the skeleton for a crossword, to accept entries for it, and even to verify and display these entries. By writing a program to do these things, we can create an 'interactive crossword', which can do a great deal more than a crossword that is merely printed in a newspaper. Besides verifying entries, it could even be made to fill in an entry if the person tackling the crossword were genuinely stuck.

For this month's programming project, we shall look at how the numbering plan for a crossword can be computed from its skeleton, and how the length of the solution to each clue can be found. We shall then go on to construct a basic interactive crossword that verifies the solver's attempt at the answer to any clue. The programs that are presented for these activities should provide a firm basis for the creation of a truly interactive crossword that is much more user friendly than a conventional one. In creating this, we can demonstrate that the computer is a much more advanced medium for supporting a crossword than the commonly used medium of paper.

Representing and displaying a crossword

A small crossword is shown, together with its clues in Figure 1. The solution to this is contained in the programs presented later on. The skeleton of the crossword is shown in Figure 2. Obviously, it consists only of black and white squares, which makes the problem of displaying it quite simple. In fact, we shall represent it with a surround of dark squares, as shown in Figure 3. By doing this, we can treat the squares at the edge of the crossword itself in the same way as all the others during computations on them. It also makes the display of the crossword on the screen more effective.

The crossword can be represented in the computer by using a two-dimensional array of string variables. Each element of the array can 'cover' a single square of the crossword, by containing the letter that should be placed in that square when the crossword is filled in, or some other character to represent a black square (we shall use a space character for this purpose).

We shall use an array named CROSS$ to store the crossword. As our crossword has 6 rows and 6 columns, we shall give CROSS$ dimensions of 7 by 7, and then rows 0 and 7, and columns 0 and 7 can be used to hold the border. We can now represent the crossword itself by:

```
10 DIM CROSS$(7,7),N(6,6)
20 FOR J=1 TO 6: FOR K=1 TO 6
30 READ CROSS$(J,K)
40 NEXT K: NEXT J
50 DATA "A"," ","I","T"," "," "
60 DATA "B","A","S","I","C"," "
70 DATA "S","S"," ","N","O","T"
80 DATA " ","C"," "," ","M","I"
90 DATA "B","I","T"," ","A","M"
100 DATA "T","I"," ","A","L","E"
```

Figure 1. Crossword with clues and solution



Clues

ACROSS
2. 1982 was the year for this
4. Simple language
7. Bends or Nazis
8. For inventing logic
10. Half a minute? No a third
11. A little information
12. Morning in Cambridge
13. Reliable chip maker
14. Real, but not variable

DOWN
1. Nothing negative about this function
2. To be the third person
3. Can there be metal in it
5. A code, as I see with one eye
6. Not Basic, not Pascal, but both
9. What you need to do crosswords
11. You can soon buy some of this

The border can be added by:

```
110 FOR L=0 TO 7
120 CROSS$(0,L)=" ": CROSS$(7,L)=" "
130 CROSS$(L,0)=" ": CROSS$(L,7)=" "
140 NEXT L
```

After this the crossword can be displayed by a simple procedure which we shall package as a subroutine starting with the line number 1000. It will display the crossword in the opposite style to that in which it appears on paper, with the squares that are black on paper appearing light on the screen. After adding the calling instruction
150 GOSUB 1000
the display subroutine is written as:

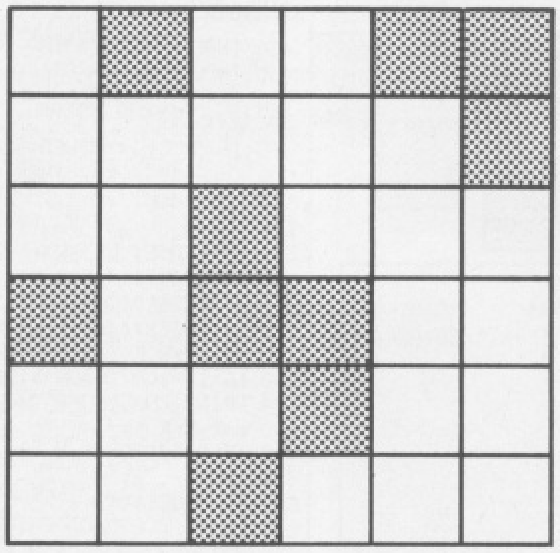


Figure 2. The crossword skeleton

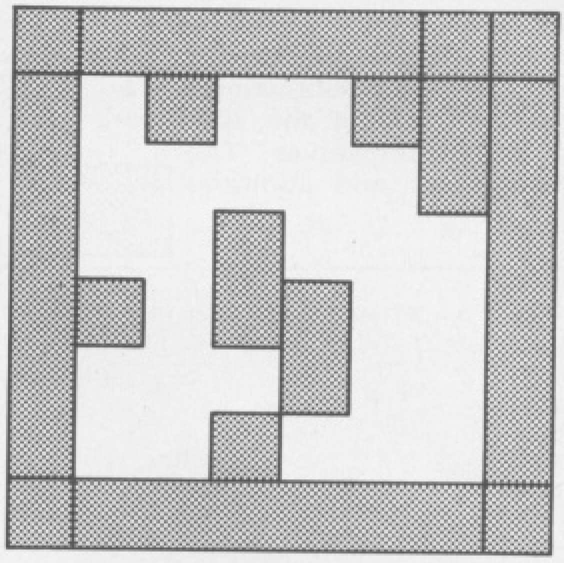


Figure 3 Crossword with surround

```
150 GOSUB 1000
160 END
410 IF CROSS$(ROW,COL)<>" " THEN PRINT CHR$(166);: GOTO 440
1000 PRINT "[illegible]"
1010 FOR J=0 TO 7: FOR K=0 TO 7
1020 IF CROSS$(J,K)=" " THEN PRINT CHR$(166);: GOTO 1040
1030 PRINT " ";
1040 NEXT K: PRINT: NEXT J
1050 RETURN
```

This is fine if we want to display the crossword, but a crossword puzzle solver wants to see the skeleton so that it can be filled in. However, our display subroutine can be adapted to give only the skeleton by changing line 1030 to:
1030 PRINT " ";
so that it prints a space rather than any letter forming part of a solution. Again, the crossword skeleton will appear on the screen with the reverse of its appearance on paper.

Finding the numbering plan

The numbering plan for a crossword is the set of numbers that is added to it to identify the positions in which the solutions of the clues are to be written. The solution to 5 across, for example, starts in the square numbered 5 and goes across, or horizontally, from there, while the solution to 7 down starts in the square numbered 7 and is written downwards from there.

The numbering plan can be determined automatically from the skeleton. This is done by examining each square in turn, row by row, starting at the top left and finishing at the bottom right. The numbers are associated only with blank squares, so the others may be ignored during this process. When each blank is examined, a code is assigned to it in a way that reflects its four neighbouring squares to the north, east, south and west. All the possible configurations for the squares surrounding a blank square are shown in Figure 4. The code is determined by counting 1 for a black square to the north, 2 for one to the east, 4 for one to the south, and 8 for one to the west. The codes are also shown on Figure 4. A little thought will show that solutions can only begin in a square having one of the codes 1,3,8,9,11,12 and 13.

But the codes give more information than just where a solution begins. They also show whether the word goes across or down, or even if a square can have answers going both across and down starting from it. By referring to Figure 4 again, we can see that words starting in a square with the code 8,12 or 13 go across, and those starting in a square coded as 1,3 or 11 go down. If a square has the code 9 then there will be words going across and down from it.

Using this information, we can find the numbering plan and print it on the skeleton with the following program, which computes the numbers and stores them in an array named N. This section of program starts at line 150, overwriting the call to the display subroutine, which is not needed again as this section of program contains its own display routine. After adding the dimensions for the array N to line 10 with:

10 DIM CROSS$(7,7),N(6,6)

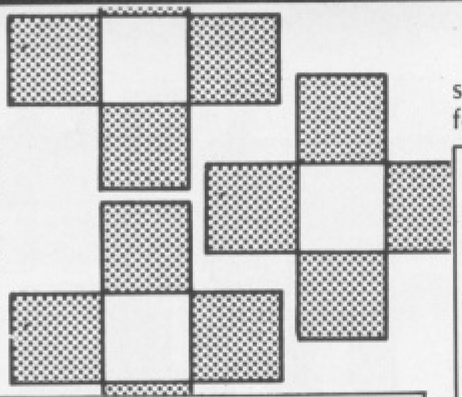
the addition to the program is:

```
10 DIM CROSS$(7,7),N(6,6)
20 FOR J=1 TO 6: FOR K=1 TO 6
30 READ CROSS$(J,K)
40 NEXT K: NEXT J
50 DATA "A"," ","I","T"," "," "
60 DATA "B","A","S","I","C"," "
70 DATA "S","S"," ","N","O","T"
80 DATA " ","C"," "," ","M","I"
90 DATA "B","I","T"," ","A","M"
100 DATA "T","I"," ","A","L","E"
110 FOR L=0 TO 7
120 CROSS$(0,L)=" ": CROSS$(7,L)=" "
130 CROSS$(L,0)=" ": CROSS$(L,7)=" "
140 NEXT L
150 N=1
160 FOR ROW=1 TO 6: FOR COL=1 TO 6
170 IF CROSS$(ROW,COL)=" " THEN 240
180 T=0
190 IF CROSS$(ROW-1,COL)=" " THEN T=T+1
200 IF CROSS$(ROW,COL+1)=" " THEN T=T+2
210 IF CROSS$(ROW+1,COL)=" " THEN T=T+4
220 IF CROSS$(ROW,COL-1)=" " THEN T=T+8
230 IF T=1OR T=3OR T=8OR T=9OR T=11OR T=12OR T=13 THEN N(ROW,COL)=N: N=N+1
240 NEXT COL
250 NEXT ROW
260 FOR ROW=0 TO 7: FOR COL=0 TO 7
270 IF CROSS$(ROW,COL)=" " THEN PRINT CHR$(166);: GOTO 300
280 IF N(ROW,COL)<>0 THEN PRINT RIGHT$(STR$(N(ROW,COL)),1);: GOTO 300
290 PRINT " ";
300 NEXT COL:PRINT: NEXT ROW

READY.
```

Note that when the skeleton is printed complete with numbers, any number from 10 upwards is truncated so that only its least significant digit is displayed. This is for the very good reason that only one character can be displayed in one character position! But despite the fact that only the last digit of the number is displayed the number is computed in full. To prove this, we can print a list of the numbers and directions for all the clues in the crossword by adding the lines
222 IF T=1 OR T=3 OR T=11 OR T=9 THEN PRINT N; "DOWN"
224 IF T=8 OR T=12 OR T=13 OR T=9THEN PRINT N;"ACROSS"

The length of each solution can be computed and added to the list of clues by amending the two lines just given so that they each call a subroutinne, and then adding the subroutines themselves. The amendments and additions are:



```
222 IF T=1OR T=3OR T=11OR T=9 THEN PRINT N;" DOWN";: GOSUB 900
224 IF T=8OR T=12OR T=13OR T=9 THEN PRINT N;" ACROSS";GOSUB 800


310 END
800 W=1: X=COL
810 X=X+1
820 IF CROSS$(ROW,X)<>" " THEN W=W+1: GOTO 810
830 PRINT "(";W;")",
840 RETURN
900 W=1: Y=ROW
910 Y=Y+1
920 IF CROSS$(Y,COL)<>" " THEN W=W+1: GOTO 910
930 PRINT "(";W;")",
940 RETURN
```

Interactive display of word

Our interactive crossword can now be programmed fairly simply to accept an attempt at a solution and to find whether it is correct or not by comparing it with the answer that it already holds. This is the point at which it really starts to become interactive.

To enter a potential solution to a clue, we could expect the user to provide us with three things: the number of the clue, whether it goes across or down, and the attempt itself. From the number we can find the starting position of the word in CROSS$ by searching the array N to find the position occupied by the number. We can then find the code associated with the startinng position to confirm that the code does indeed go across or down, so verifying the second item entered by the user. Finally, since we know how to find the length of the word starting at any valid position, we can find what the answer really is, by extracting it from CROOS$, and compare it with the entered attempt to see whether it is correct or not. All the necessary actions have been programmed previously, although not in exactly the way that we now need. By adapting the previous program segments, we can arrive at the following:

```
450 INPUT "NUMBER";M
460 INPUT "A FOR ACROSS, D FOR DOWN";D$
470 INPUT "SOLUTION"; S$
480 J=1: K=1
490 IF N(J,K)=M THEN 520
500 K=K+1: IF K=7 THEN K=1: J=J+1
510 GOTO 490
520 T=0
530 IF CROSS$(J-1,K)=" " THEN T=T+1
540 IF CROSS$(J,K+1)=" " THEN T=T+2
550 IF CROSS$(J+1,K)=" " THEN T=T+4
560 IF CROSS$(J,K-1)=" " THEN T=T+8
570 IF (T=8OR T=9OR T=12OR T=13) AND D$="A" THEN 600
580 IF (T=1OR T=3OR T=9OR T=11) AND D$="D" THEN 650
590 PRINT "NO CLUE ";M;D$: STOP
600 W=1 X=K
610 IF MID$(WORD$,W,1)<>CROSS$(J,X) THEN PRINT "INCORRECT": STOP
620 X=X+1
630 IF CROSS$(J,K)<>" " THEN W=W+1:GOTO 610
640 PRINT "CORRECT": STOP
650 W=1:Y=J
660 IF MID$(WORD$,W,1)<>CROSS$(Y,K) THEN PRINT "INCORRECT":STOP
670 Y=Y+1
680 IF CROSS$(Y,K)<>" " THEN W=W+1:GOTO 660
690 PRINT "CORRECT": STOP
700 END


READY.
```

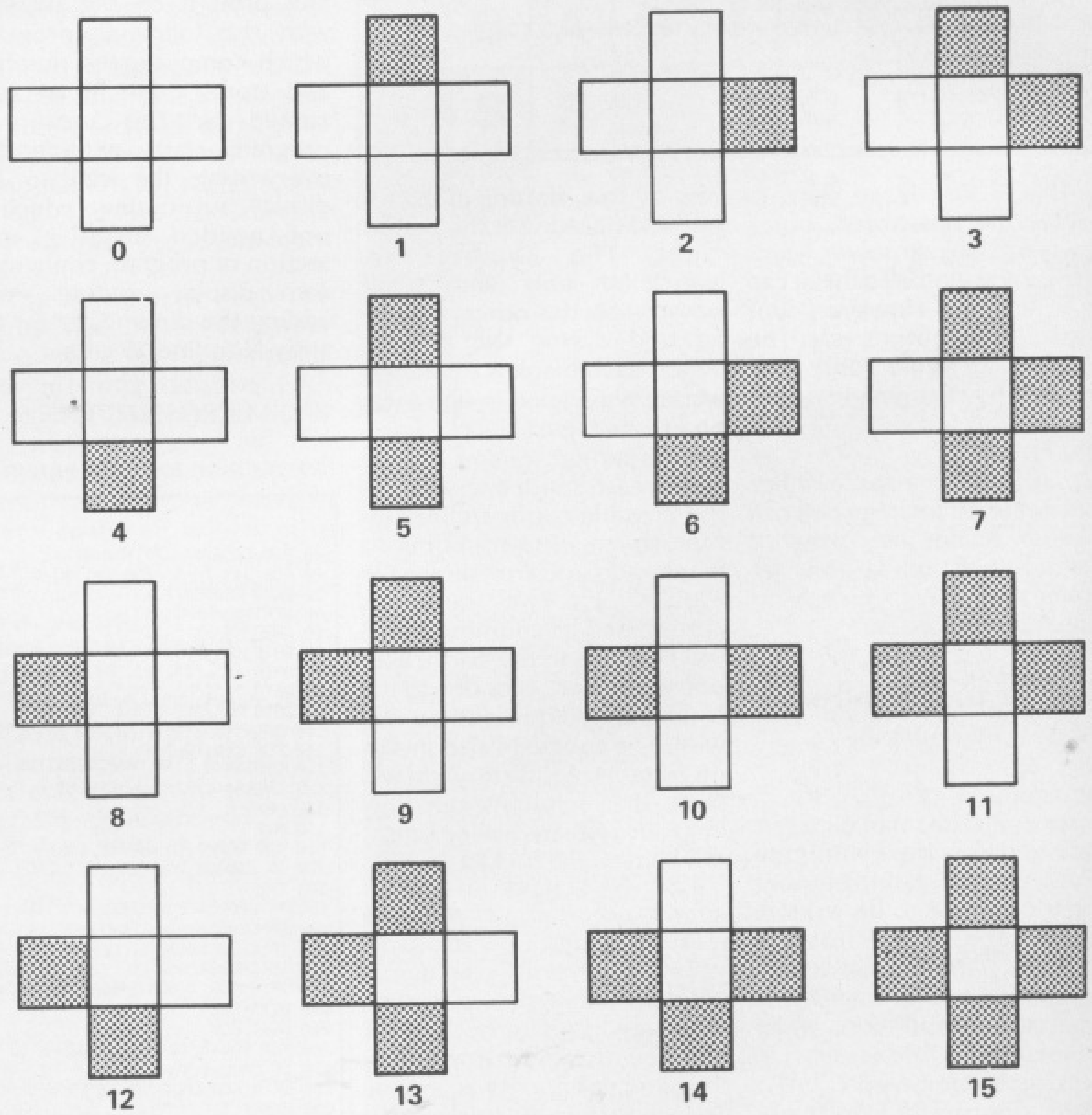
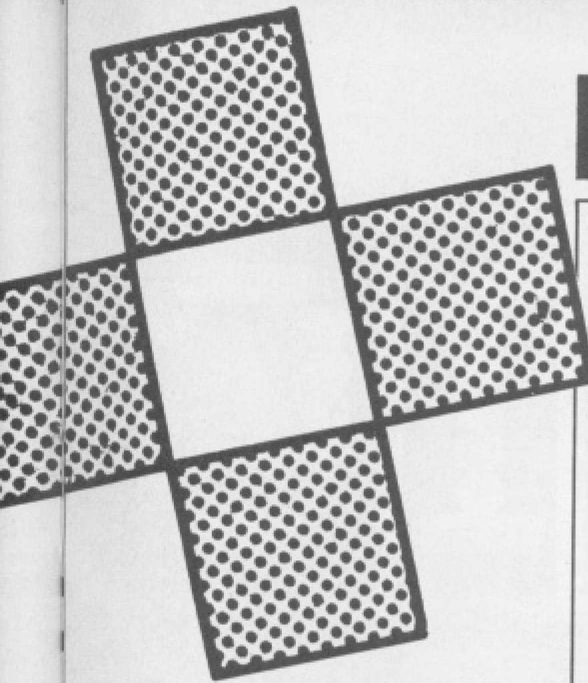


Figure 4. Blank squares and their codes



Note that by replacing STOP instructions wherever they appear (that is, in lines 590, 610, 640, 660 and 690) by GOTO 450 the program can be made to accept entries continually. If this is done, it will also be necessary to take steps to tidy up the display. It can also be observed that the subroutines starting at lines 800 and 900 have been written in different ways. They perform essentially the same functions, but the ways in which they are written show that the same actions may be programmed in very different ways.

Summary

The program fragments presented here provide a kit of parts for the construction of a basic 'interactive crossword'. The parts need to be assembled coherently, the code needs to be made secure against the entry of invalid data, and there is scope for further development before a fully interactive crossword program is produced.

Nevertheless, many of the necessary building blocks and ideas are here. The basic structure of the program needs improving, not least by packaging routines that are needed more than once as subroutines. The ultimate success of a program in providing an interactive crossword may depend on its structure.

A complete listing of the program developed in this article is given in Figure 5.

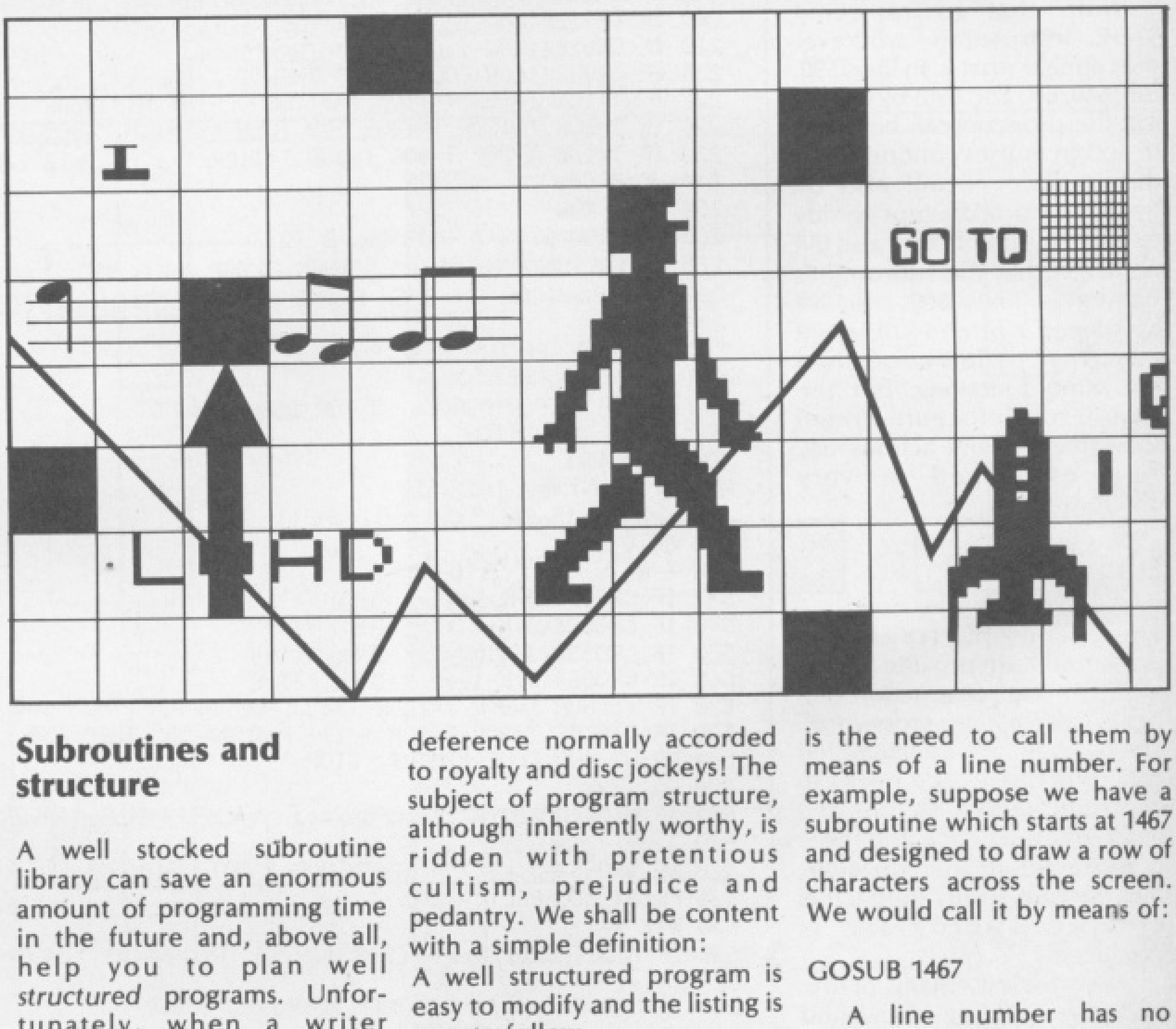
Program Listing

```
10 DIM CROSS$(7,7),N(6,6)
20 FOR J=1 TO 6: FOR K=1 TO 6
30 READ CROSS$(J,K)
40 NEXT K: NEXT J
50 DATA "A"," ","I","T"," "," "
60 DATA "B","A","S","I","C"," "
70 DATA "S","S"," ","N","O","T"
80 DATA " ","C"," "," ","M","I"
90 DATA "B","I","T"," ","A","M"
100 DATA "T","I"," ","A","L","E"
110 FOR L=0 TO 7
120 CROSS$(0,L)=" ": CROSS$(7,L)=" "
130 CROSS$(L,0)=" ": CROSS$(L,7)=" "
140 NEXT L
150 N=1
160 FOR ROW=1 TO 6: FOR COL=1 TO 6
170 IF CROSS$(ROW,COL)=" " THEN 240
180 T=0
190 IF CROSS$(ROW-1,COL)=" " THEN T=T+1
200 IF CROSS$(ROW,COL+1)=" " THEN T=T+2
210 IF CROSS$(ROW+1,COL)=" " THEN T=T+4
220 IF CROSS$(ROW,COL-1)=" " THEN T=T+8
222 IF T=1OR T=3OR T=11OR T=9 THEN PRINT N;" DOWN";: GOSUB 900
224 IF T=8OR T=12OR T=13OR T=9 THEN PRINT N;" ACROSS";: GOSUB 800
230 IF T=1OR T=3OR T=8OR T=9OR T=11OR T=12OR T=13 THEN N(ROW,COL)=N: N=N+1
240 NEXT COL
250 NEXT ROW
260 FOR ROW=0 TO 7: FOR COL=0 TO 7
270 IF CROSS$(ROW,COL)=" " THEN PRINT CHR$(166);: GOTO 300
280 IF N(ROW,COL)<>0 THEN PRINT RIGHT$(STR$(N(ROW,COL)),1);: GOTO 300
290 PRINT " ";
300 NEXT COL:PRINT: NEXT ROW
450 INPUT "NUMBER",M
460 INPUT "A FOR ACROSS, D FOR DOWN";D$
470 INPUT "SOLUTION"; S$
480 J=1: K=1
490 IF N(J,K)=M THEN 520
500 K=K+1: IF K=7 THEN K=1: J=J+1
510 GOTO 490
520 T=0
530 IF CROSS$(J-1,K)=" " THEN T=T+1
540 IF CROSS$(J,K+1)=" " THEN T=T+2
550 IF CROSS$(J+1,K)=" " THEN T=T+4
560 IF CROSS$(J,K-1)=" " THEN T=T+8
570 IF (T=8OR T=9OR T=12OR T=13) AND D$="A" THEN 600
580 IF (T=1OR T=3OR T=9OR T=11) AND D$="D" THEN 650
590 PRINT "NO CLUE ";M;D$: STOP
600 W=1:X=K
610 IF MID$(WORD$,W,1)<>CROSS$(J,X) THEN PRINT "INCORRECT":STOP
620 X=X+1
630 IF CROSS$(J,K)<>" " THEN W=W+1:GOTO 610
640 PRINT "CORRECT": STOP
650 W=1:Y=J
660 IF MID$(WORD$,W,1)<>CROSS$(Y,K) THEN PRINT "INCORRECT":STOP
670 Y=Y+1
680 IF CROSS$(Y,K)<>" " THEN W=W+1:GOTO 660
690 PRINT "CORRECT": STOP
700 END
800 W=1: X=COL
810 X=X+1
820 IF CROSS$(ROW,X)<>" " THEN W=W+1: GOTO 810
830 PRINT "("; W; ")",
840 RETURN
900 W=1: Y=ROW
910 Y=Y+1
920 IF CROSS$(Y,COL)<>" " THEN W=W+1: GOTO 910
930 PRINT "("; W; ")",
940 RETURN
```

Figure 5 The complete listing

THE BASIC FACTS PT 6

Subroutines and User Defined Graphics are the subjects under discussion in this month's instalment of our BASIC series from A.P. and D.J. Stephenson.



FROM A SUPERFICIAL viewpoint, we could define a subroutine as a collection of programming lines terminating in the keyword RETURN and activated (called) by the keyword GOSUB. After a few weeks playing around with programs we would probably reach the conclusion that a subroutine functions as a kind of *subcontract* to the main program. Like a subcontractor in the building trade who specialises in, say, making window frames, a subroutine can be given the responsibility of providing a picturesque and coloured border around the screen or, on a slighte higher plane, finding the two solutions of a quadratic equation. The benefit of subcontracting in real life is the fact that advantage can be taken of specialised expertise and equipment. It is the same with programminng. As your experience widens, you will begin to notice that programs, however complex and different in overall objectives, contain many similar ingredients even though there may be differences in variable names. Once this is recognised, you will realise that well designed subroutines can be used over and over again in a wide variety of programs. As a result, your approach to programming could change dramatically. In fact you will probably ease off writing complete programs until you have built up a stock of useful, general purpose subroutines — a subroutine library in fact.

Subroutines and structure

A well stocked subroutine library can save an enormous amount of programming time in the future and, above all, help you to plan well *structured* programs. Unfortunately, when a writer introduces the word 'structure' it usually means that the next few thousand words will be devoted to a boring explanation of what it means and, worse still, a certain amount of name dropping. Names like Edsger W. Dijkstra (reputedly quite clever but something of an intellectual snob) and Niklaus Wirth (the creator of PASCAL) are mentioned with the kind of humility and deference normally accorded to royalty and disc jockeys! The subject of program structure, although inherently worthy, is ridden with pretentious cultism, prejudice and pedantry. We shall be content with a simple definition:

A well structured program is easy to modify and the listing is easy to follow.

The liberal use of subroutines within a program will certainly contribute to the structure providing they are reasonably well thought out in the first place.

Subroutine layout

Although subroutines are good for structure, there is no denying that their worst aspect is the need to call them by means of a line number. For example, suppose we have a subroutine which starts at 1467 and designed to draw a row of characters across the screen. We would call it by means of:

GOSUB 1467

A line number has no humanity. It is abstract symbolism and, even worse, it is quite probable that the number is provisional and will most likely be changed as the result of a renumbering excercise during program development.

How much better it would be if BASIC allowed us to choose a meaningful label instead of a line number. For example, instead of GOSUB

1467, it would be much easier to follow the listing if we could write GOSUB DL (aabbreviation for Draw Line). Alas, we are not allowed to, so the next best thing is to make sure that all subroutines stand out well on a listing by a REM statement which briefly describes their function.

For example, the following few lines are a guide to how a 'draw line' subroutine should appear on a listing:

```
9999  REM DRAW LINE SR
10000 PRINT"----------------"
10010 RETURN
```

This is not expected to score high marks for subtlety or invoke Dijstra's envy but it serves to illustrate two points:

(a) The REM statement is given a 'one-less' line number.
(b) The first effective line is a nice round figure in thousands.

Neither of these points are mandatory but they help to keep a program looking tidy. The reason for choosing an odd number for the REM line is to emphasise the fact that it is an 'outsider' (non executable) and *not* the starting number for the subroutine. The rule is never, never GOSUB (or GOTO for that matter) to a REM statement because, to save memory, you may eventually be tempted to cut them out from your working copy. If you call the above with GOSUB 10000, the REM can be removed at any time without fear of risking an error message from the interpreter.

Still on the subject of line numbers, it is a good plan to number all subroutines in a program starting at *round thousands*. For example, the first subroutine at line 10000, the next at line 11000 and so on. This will obviously leave masses of unused line numbers in between but who cares? The program will be easier to follow so, according to our definition, it is a worthwhile dodge because it contributes to our simplified definition of structure.

Keyboard input subroutine

Perhaps the most commonly required subroutine is one which validates keyboard input. When string data is requested from the keyboard in response to an INPUT prompt it is possible that the operator might hit RETURN before the data is entered. The input is therefore a 'null string' which can be infuriating unless some trap can be laid to prevent it. There may also be a limit on the number of characters which can be entered. To save writing these traps every time an INPUT statement appears, it can be solved once and for all by enclosing the lot, including the INPUT statement, within a subroutine. For example:

```
11999 REM INPUT VALIDATION SR
12000 K$="":INPUT K$
12010 IF K$="" THEN 12000
12020 IF LEN(K$)    L THEN PRINT"TOO LONG":
GOTO 12000
12030 RETURN
```

Note that K$ holds the keyboard response and that the number of characters allowed must be assigned to L before calling. A typical calling sequence would be:

```
100 PRINT"ENTER NAME OF ORGANISM":L=18
110 GOSUB 12000
120 N$=K$
```

Note that the number of characters has been limited to 18 (an arbitrary whim of the programmer) and, on returning from the subroutine, the general purpose variable, K$, is re-assigned to N$.

It is conventional, but not mandatory, to shove all subroutines down to the *bottom of the program.*

Subroutine material

In the early days of computing, memory was expensive — a 1K magnetic core memory would cost around £1000. Consequently, the foremost consideration for the programmer was to ensure that every precious byte earned its keep. As a direct result, the attitude towards subroutine usage was much different to what it is now. A subroutine was used primarily to avoid *repetition* programming. If a particular set of lines was going to be used several times in the same program then it was sound economics to bundle them up into the form of a subroutine which could be called whenever needed.

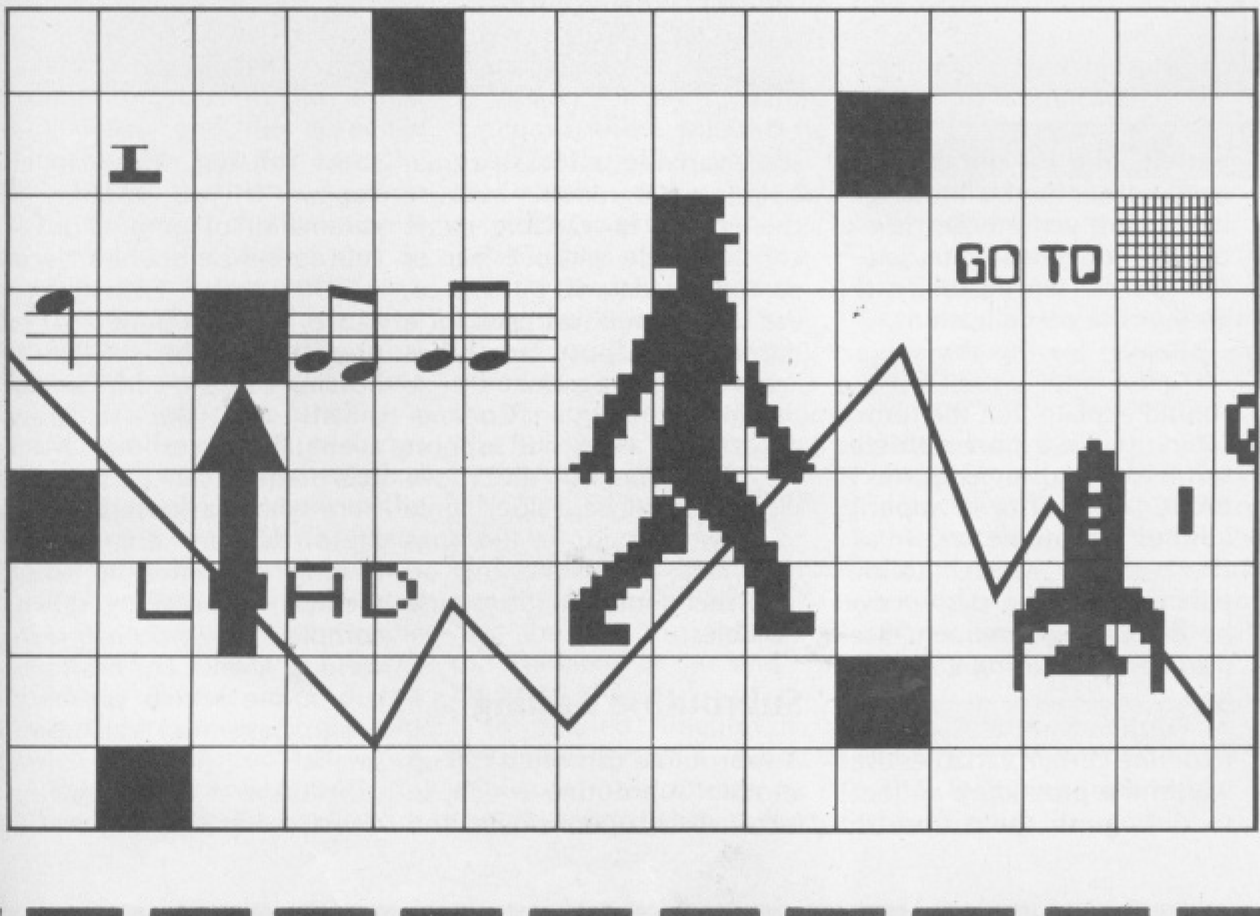
Nowadays, the position is different. Memory is relatively cheap so saving it is not always an overriding consideration. The criterion for inclusion into a subroutine is whether or not the function it performs can be recognised as a 'logical entity'. Even if it is to be called only once in the program the function can still qualify for subroutine status.

Our 'draw line' subroutine above, although apparently trivial, is certainly a candidate for a subroutine. It is a logical entity (has a clearly defined single function) and, in addition, will probably be required several times during a program RUN.

In fact, the modern 'program' is often a relatively short, skimpy affair, consisting of little more than a series of subroutine calls. Using the analogy we made at the beginning, the main contractor does very little, preferring to sit on his/her backside and farm out most of the work to subcontractors.

Passing parameters

Some subroutines are



complete in themselves and require no information or help whatsoever from the calling program. The 'draw line' subroutine provides such an example. All we have to do is call it and it prints out a string '-' characters to form a dashed line.

But there may be times when we want to draw lines using other characters. For example, a line of '★' or perhaps '+'. Do we then use another subroutine employing a different character? We could, of course, but it would be a shocking waste of programming energy and memory. The more efficient way would be to re-organise the subroutine so that it can draw a line using any character we choose. This will entail substituting the literal character '-' with a string variable and using a FOR/NEXT loop to print it out a number of times.

For example:

```
9999  REM DRAW LINE
10000 FOR K = 1 TO 20
10010 PRINT L$;
10020 NEXT
10030 RETURN
```

This will print out a row of twenty characters, the actual character being that which happens to be in L$ at the time the subroutine is called.

This requirement highlights the problem of 'parameter passing' because the subroutine is no longer an independent animal. We must ensure that when we call it, the character we intend to use is assigned to the variable L$. In technical jargon, we must pass the character parameter. For example, if the line is too be drawn with '★', the calling procedure will now be:

```
100 L$ = "★"
110 GOSUB 10000
```

Thus, in return for a little extra complication in the subroutine, we enjoy the facility of using the same subroutine for drawing a row of any character we choose and, what's more, the characters can be different each time it is called. It is worth mentioning here that instead of assigning L$ to the literal character '★', we could have used the form:

```
100 L$ = CHR$(x)
```

where X is the character code — the code for '★' is CHR$(42). It is possible to increase the generality of the subroutine even more by arranging for the end-of-loop counter in line 10000 to be passed as a parameter. For example:

```
10000 FOR K = 1 TO L
```

However, this means that two parameters must now be passed, one for the character and another for the line length, illustrating yet another law of our friend (?) Septimius Sod — the greater the flexibility, the greater the complication.

Before leaving the subject of parameter passing, we should explain that the term is often used in a more restricted form. More advanced forms of BASIC now offer a superior kind of subroutine known as a *Procedure* which allows parameters to be passed over by the calling statement itself instead of requiring a separate line.

Furthermore, it is possible to define certain variables used within the procedure as 'local' to distinguish them from the 'global' variety. These terms need some explanation. When a variable, say X, within a subroutine is declared to be 'local', it can be used freely without fear of corrupting important data it may have acquired outside the boundaries of the subroutine. In other words, the global value of X is preserved even though its local value may be varied by the subroutine. For example, if X = 4 before calling and the subroutine alters it to 24, the 4 is automatically restored to X again after RETURN. The facility to declare some variables as local is a great help to a programmer because the choice of variable name can be made without fear of corrupting data if, by chance, the same name was used for a different purpose in other areas of the program.

Unfortunately, Commodore BASIC does not support local variables but the above discussion still has value, if only to draw attention to the bugs that can arise following an incorrect choice of subroutine variables.

Subroutine nesting

A subroutine can often call up another subroutine which, in turn, can call upon another and so on. This technique, known as *nesting* is illustrated in Figure 6.1

There is a limit to the number of subroutines which can be nested because the interpreter has to store all the *return addresses* in a reserved and restricted area in RAMA known as the *stack*. The stack is organised as a LIFO memory, (Last In First Out). Although the BASIC programmer is blissfully unaware of LIFO action, the sudden appearance of the message 'OUT OF MEMORY' can appear on the screen even when there is plent of usable RAM still left. This can also happen if you commit the cardinal sin of jumping out of a subroutine before the normal RETURN route. Each time we cause a subroutine to exit prematurely, the stack is left holding a return address which means that the stack will eventually overflow if the subroutine is called many times from within a loop. Providing these dangers are avoided, nesting provides a useful method of breaking down a complex subroutine into various 'levels'. For example, a subroutine which presents a Menu page may call upon a smaller (lower level) subroutine to draw a demarcation line between the

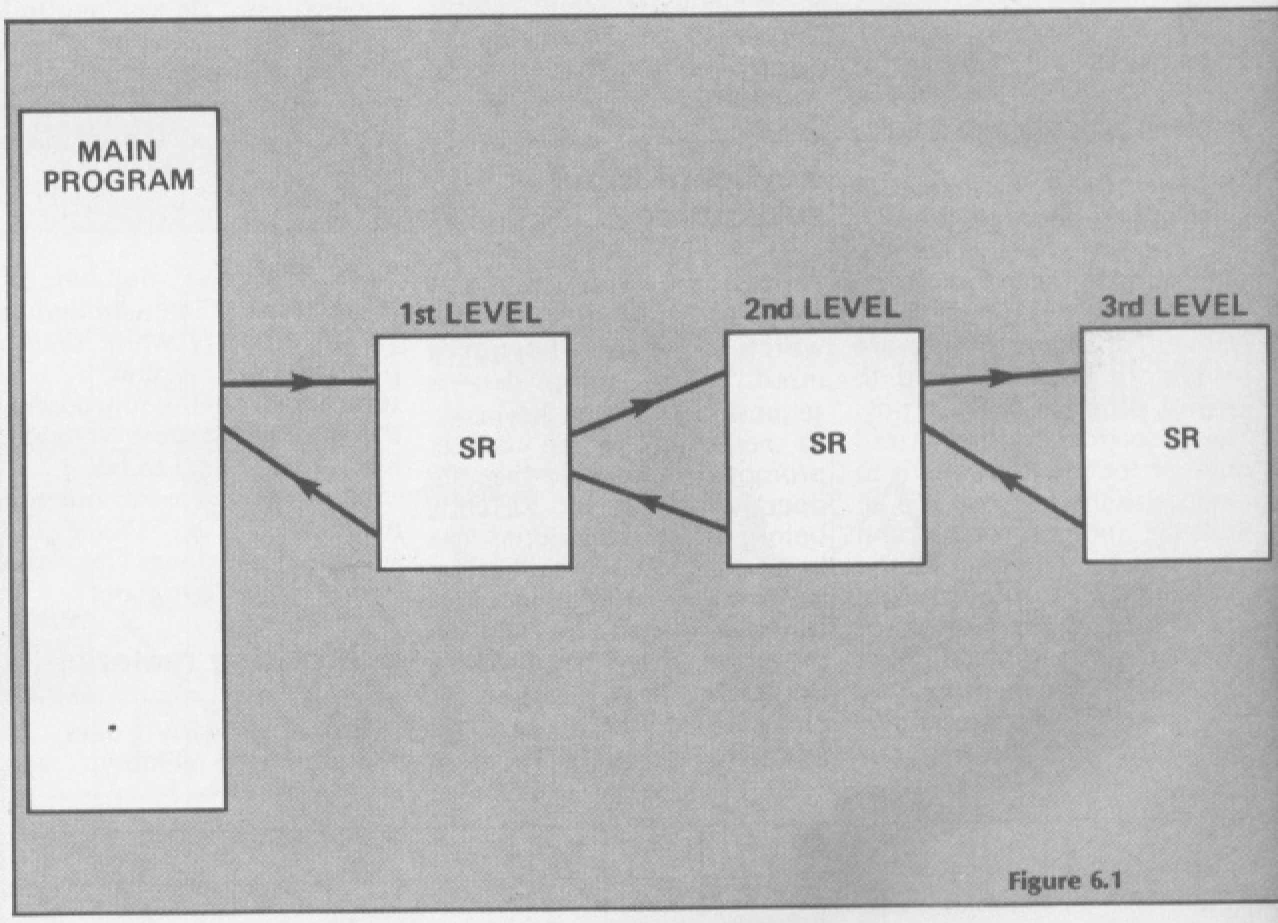


Figure 6.1

heading and the start of the menu options. It may be called a second time to separate the bottom of the menu from the typical prompt, 'Enter option required'.

The ON GOSUB statement

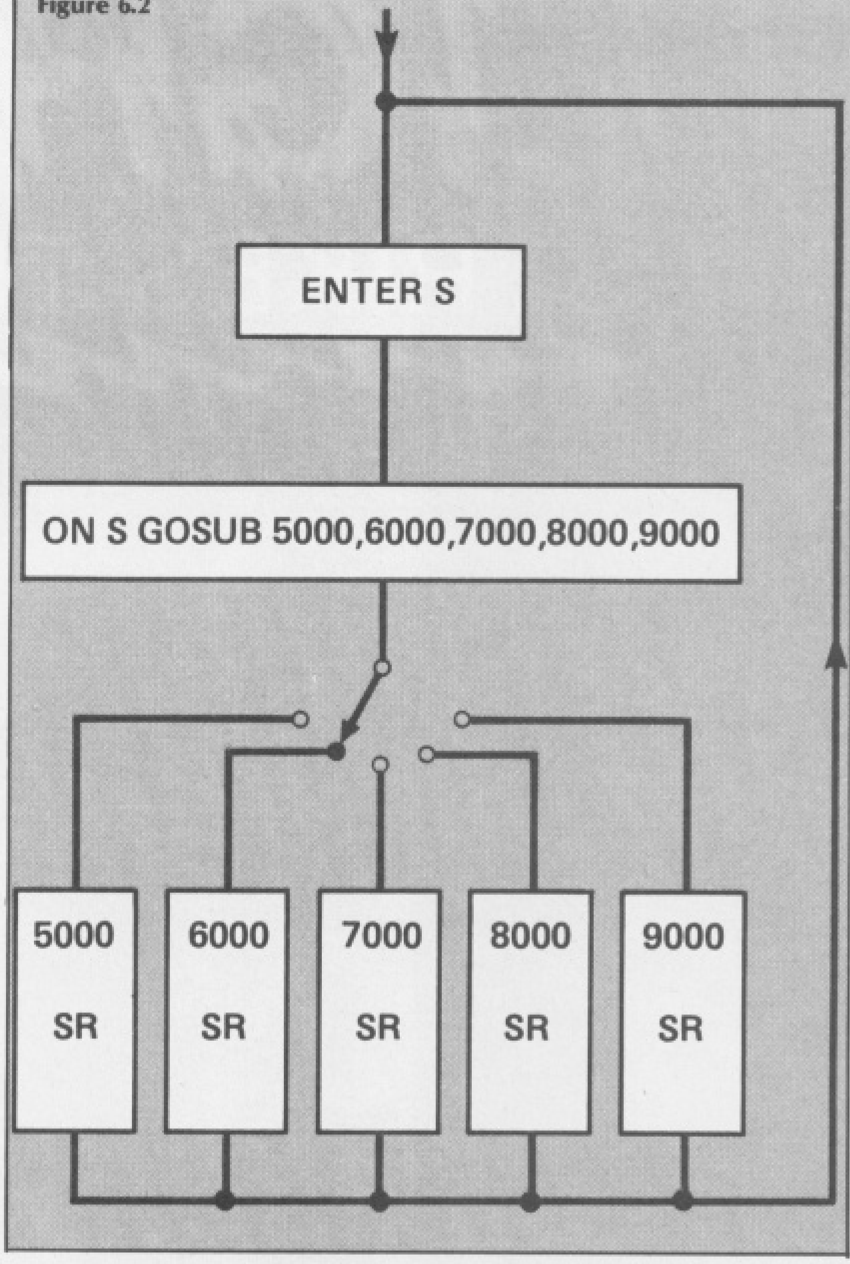
The mention of menu options is a cue for introducing the ON GOSUB statement. The following example will serve to illustrate the syntax:

ON S GOSUB 5000,6000,7000, 8000,9000

If S = 1, the subroutine at line 5000 is called, if S = 2, the subroutine at line 6000 is called and if S = 5 it will call on the last subroutine at line 9000. Those who have dabbled in electrical circuits will recognise this as the software equivalent of the *single pole, multiway switch* as shown in Figure 6.2

Assuming a separate subroutine is responsible for handling each option, the actual program can be reduced to a simple affair. It need only contain a few assignment lines for setting the initial conditions and presenting the menu — the subroutines can be left to do all the work. We could, of course, go a stage further and make the actual menu page the subject of a subroutine.

Figure 6.2



User defined functions

A 'function' is a term used with a variety of meanings, depending on both the context in which and the academic level of the text. For example, in higher mathematics, even the definition of a function is usually good for twenty or so pages of mind boggling text. Provisionally, we will describe a function as something that does something to something else! For example, SIN(X) is a function because it performs that particular mathematical operation on X. We input the value of X to the SIN(X) 'black box' and it emerges with a totally different value after it has been messed around by the function. SIN(X), COS(X), TAN(X), EXP(X) and a few others, are some of the oft-used standard functions which are available to us in BASIC. There are hundreds of other functions which could also come under the heading of 'standard' but for obvious reasons, BASIC can not supply them all.

Apart from standard functions, there will also be a need for functions peculiar to the needs of a particular program. To satisfy such needs, BASIC gives us the means of writing our own functions bny using the keyword DEF FN A (X), where X (or indeed any other legitimate character) is the variable to be acted upon. The full syntax is as follows:

```
DEF FN name (X) = some equation containing X
```

For example, DEF FN G (X) = X+5: the function name is G, the variable is X and the equation is X+5.

A defined function is, in some respects, a kind of miniature subroutine so it shoudl follow that a formal method must exist for 'calling' it. This is done by using FN(C), where C is the actual value to be used in the function. All this sounds very confusing so an example is indicated:

```
100 DEF FN G (X) = 2★X+X ^3
```

Some time later in the program we might want to evaluate this equation and print out the result when X = 3. This can be achieved by the following line:

```
400 PRINT FN G (3)
```

This would evaluate 2★3+3 ^3 and print out the result, 33. We don't always want to print out the function. For example, we couild use FN as an ordinary variable in part of another expression as in the following example:

```
600 A = G + FN G (5)
```

The following rule must be observed:

The function must be defined before it is called, i.e, DEF FN must come before FN.

Once a function has been defined, you can call on it as many times as you wish within the same program and use different values of the variable each time. The function can be complex and contain other functions such as:

```
DEF FN S (Z) = SIN(Z) + COS(Z)
```

S is the function name, Z is the variable

```
DEF FN D4 (PF) = LOG(PF) ★ EXP(PF)
```

D4 is the function name, PF is the variable.

The equation can contain additional variables other than the functionn variable providing, of course, they have been previously assigned. For example:

```
DEF FN G (X) = X ^2 + K
```

It is also allowable to use a variable, instead of a constant, when calling with FN providing it has previously been assigned a value. For example:

```
500 FN S (G)
```

We have suggested that a defined function can be thought of as a miniature subroutine but it is time we pointed out the differences between them.

1. A subroutine can occupy as many lines required. A defined function can occupy only one computer line.
2. A subroutine can contain strings or numeric variables. A defined function can only handle numerics in Commodore BASIC.
3. It doesn't matter where a subroutine is situated because it can be called from an earlier or later line number. The function, on the other hand, must be defined by DEF FN before it can be called with FN.
4. Unlike subroutines, a defined function passes a parameter directly by the FN call instead of requiring a separate assignment line.

The main use of a defined function is to avoid writing out lengthy equations each time they are required.

LIGHTNI

STRIKES THRICE

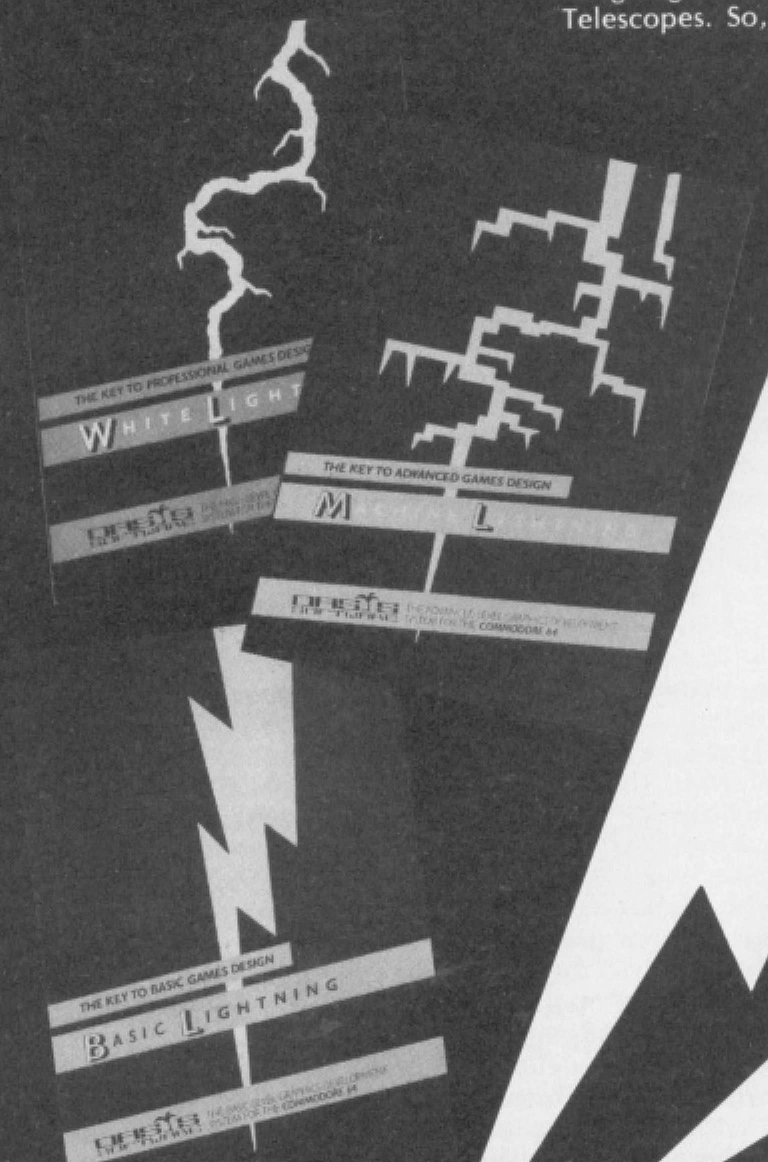
Oasis send 3 bolts from the blue. Phil South considers all three as direct hits.

WHAT IS FAST, POWERFUL, LOUD, colourful and frighteningly complex? A bolt of fork lightning? A molecule of DNA doing the 100 yard dash? No, Oasis Software have just answered this question by writing three marvellous new 'software development' packages. I know, those two words strike fear into the heart of most home computer users. But you can relax, these packages are friendly, easy to use and require only a CBM 64 (not a SAGE IV) to produce professional quality, "stand alone" computer games. They differ from games designers in that not only do they produce programs that run without the Lightning software in residence in your, or anybody's, computer, but they don't override the existing functions of the resident BASIC. (Such as they are...Oops!)

Basic Lightning

The first of the three is **Basic Lightning**, an extended, multi-tasking BASIC with a penchant for arcade games. Basic Lightning is what you might call a Structured BASIC, this means it has commands in common with other such BASIC's, like BBC BASIC and Sinclair QL SuperBASIC; IF-THEN-ELSE, REPEAT-UNTIL, CASE, PROCedures, and so on. It also has some natty routines of its own like DLOAD and DSAVE for LOAD/SAVE from/to disc — a trifle easier to type than LOAD "whatever", 8, 1, and coming a lot easier to recent converts to the 64's obvious charms. DIR is another nice touch to Basic Lightning, listing the directory of the disc in the drive (no more $ hunting). It also has commands like WHILE-WEND and CIR-CELSE-CEND, the former being a looping type function like IF-THEN and the latter being a 'multiple line' equivalent. DISABLE nukes the RUN-STOP key so some snooper can't break into your program and read it; a handy piece of built-in protection if ever I typed one! Another Basic Lightning plus, is 'windowing' where you can throw a smaller, inset, piece of graphics screen into a screen of text, or indeed onto another graphic, like a sprite, for instance. In case you didn't know, this is quite useful because the 64 can't usually display text and hi-res graphics on the same screen; it's normally rather like the hi-res screen is pulled down over the text screen.

I tell you what, the one thing that always really gets my goat about CBM BASIC, is having to remember those POKEs for background and foreground colours in the display file. Basic Lightning skirts this issue nicely with the commands TBORDER, HBORDER, HPAPER, TINK, HINK, etc. The 'T' prefix specifying a text colour and the 'H' a hi-res colour. What about colour numbers, you know, 0-15 and all that? Try BLACK, BROWN or RED, as they are the replacements in Basic Lightning. You can also shuffle the attributes around on their own in rectangular blocks, using the MOVATT or SWAPATT commands. In any arcade-style game, collision detection is a must. What's that then? That is when an alien sits around long enough to bump into one of your bullets; the alien is detecting the collision and signalling the appropriate action, in this case, erasing the alien and printing up an explosion. BA-BOOM! Heh-heh. Basic Lightning collision detection, with the DTCTON and DTCTOFF commands. Essentially, (I nearly said basically) Basic Lightning is a kind of testbed program. It starts you off before you use the other two in the series, so you can run your program in extended BASIC using the pre-formed sprite cobbling routines, and then translate it into one of the other two, and sell it as super fast machine code.



White Lightning

White Lightning is an implementation of the language FORTH. In the 1960's Charles H. Moore, an astronomer at the Kitt Peak Observatory in the USA, needed a specialist Input/Output computer language for controlling Radio Telescopes. So, being no slouch at the

NG

computer keyboard he sat down and hacked out his very own language, which he called FORTH.

White Lightning is an accurate rendering of FORTH-79, which was until last month the most recent version. It uses pre-defined words like IF-ELSE, DO-UNTIL, DO-LOOP, etc., but its real strength, and the source of all the fuss is in its definable words, an infinite number of them because they designed by you. This is done by 'colon definitions' in the following format:-

: new word old word old word 2...etc...last word ;

defining new words in terms of existing words on the 'word list', the stack of words supplied with the FORTH language. This all means that FORTH grows with your application, becoming tailored to the individual needs of the job-in-hand and becoming an expert on whatever your task is. The speed of execution of a FORTH program is close to that of machine code, but its word structure is as easy to learn as BASIC.

Apart from anything else though, the way you have to program in FORTH forces you into good programming habits...or your programs don't work. Its style is a bit like structured BASIC in the way you write a main routine and use that to control a lot of smaller routines to do the job. FORTH programs are unlike BASIC in another, too. BASIC, including Basic Lightning, is an 'interpreted' language; this means that the programs are a 'source' code which is read by a chip called the BASIC interpreter and directly executed. What happens to FORTH source code is that it is 'compiled' into a series of chunks of machine code, and executed when the user types in one of the new defined words in the source code. White Lightning source code is entered into the computer, like a text file, onto screens or pages within memory, which the compiler then reads starting at the first page.

As a supplement to the package, Oasis have included a copy of Basic Lightning and a thing called IDEAL, a sprite handling expert sub-language. IDEAL deals with sprite juggling, stretching and positioning on the screen; it also has a lot to do with important things like collision detection etc. White Lightning can create 'stand alone' programs that you can sell without restriction (I should think so too!). All Oasis want out of the deal is a small mention on the packaging of your game. Not a lot to ask if you're earning as much as Jeff Minter, Matthew Smith et al.

Machine Lightning

Lastly, there is **Machine Lightning.** The most difficult of the three but, by the Phil South Inverse Difficulty Theorum, proportionally more powerful than the other two. It is a full function 6502 processor Macro-assembler, dissembler, monitor and tracer, with Basic Lightning again, IDEAL and a gaggle of sprite/graphics added to make it interesting. I won't bother to try to explain machine code here, (A.P. & D.J. Stephenson get paid to do that) but suffice to say it's not for beginners. It's highly recommended that before you tackle Machine Lightning you buy Basic Lightning, then graduate to White Lightning. Use them and then buy a book with a title something like, 6502 Machine code for the absolute screaming novice or similar, read it, and then dive into the wonders of Machine Lightning. Agreed, Machine Lightning is one of the most user-friendly assemblers about, but you still have to know your LDA from your STA in order to make a noisy standing-on-it's-own-two-legs arcade game. For more experienced programmers, though, Machine Lightning constitutes the most complete software development package on this machine. It is 'a joy to use' (I hate that phrase, but it's true) and it is a slick, all-in-one masterpiece.

The things that are common to all the packages is their handling of sound, multi-tasking and windows. Multi-tasking, besides being an impressive piece of jargon, is like time-sharing used to be on the DEC System 10's and other mainframes. Several functions can be controlled all at once, foreground and background, each one operation for a mere 1/20th of a second at a time, inching along. If you've got 7 things happening at once, thats only 7/20th sec. to perform one step, 2½ times each function, 20 steps every second. Tasks can be assigned priority and even halted while other functions take over. Control of sounds and music is made much easier too, with parameters governing volume, frequency, attack, decay, sustain, release, waveform, filtering, ring modulation and the 'voices', the three channels.

In summing up, I have really enjoyed these packages. All three make full use of the machine's functions, enabling even a complete beginner to produce aliens, sound effects and music with style and speed. The arcade sprite library for example is choc-a-block with handy sprites to start you off. All the frogs, bug-eyed monsters, robots, tanks, flying saucers you could ever need are filed away on the cassette or disc for you to use in your own programs. The goals of the programmers were 1) that the programs be powerful, and 2) that they be easy to use. They are both of these things and more, and I think the lads at Oasis deserve a slap on the back and all the fame and fortune I feel sure they're going to get. I love the programs and will use them forever, but above all I must praise the documentation. They are the most readable and well set out manuals I have had the pleasure to review. There, I've said it; bees knees or what? Right, off you go, and the first person to come out with a Frankie Goes To Hollywood video game will have me to answer to! (*Too late. See last month's Data Statements – Ed.*)

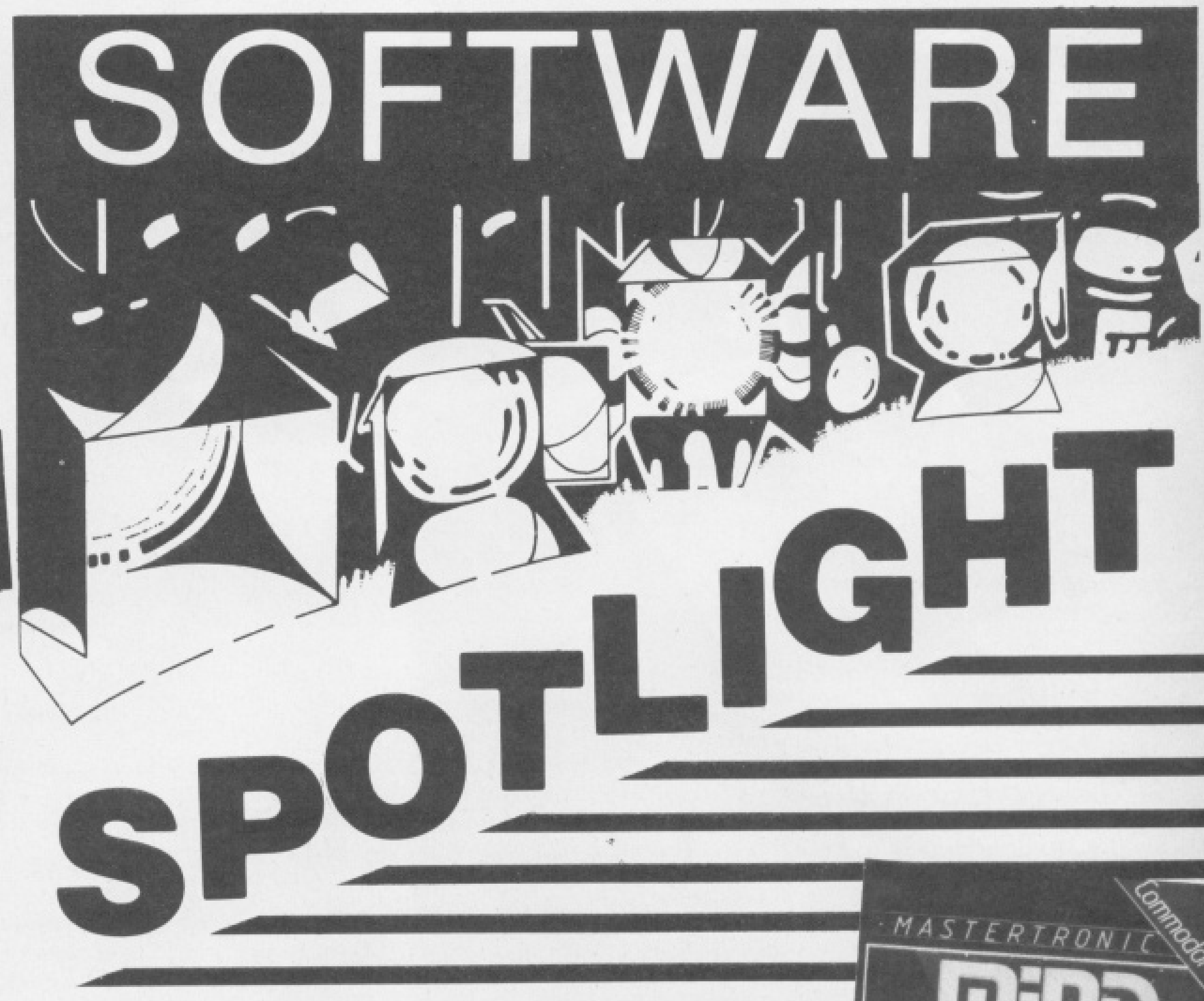
Basic Lightning costs £14.95 tape/£19.95 disc

White Lightning costs £19.95 tape/£29.95 disc

Machine Lightning costs £29.95 tape/£39.95 disc

Oasis Software, Alexandra Parade, Weston super Mare, Avon BS23 1QT.

SOFTWARE SPOTLIGHT



Software Spotlight is even bigger than ever in this great games issue.

Mr. Robot
★ ★ ★ ★
Beyond-Datamost
£8.95
CBM 64 + Joystick

I AM VERY TEMPTED TO GIVE this game full marks for quality. The numerous screens (22) of play in this game are easily accessible throughout, by pressing F3 any start screen can be selected from the easiest to the hardest. The graphics in the game are verging on being very good with the inclusion of sprites for the main character and some of his adversaries.

The main idea of this game is to guide your robot through the 22 screens, whilst avoiding the alien fireballs and collecting the power pills. The catch here is that if you do not collect all the pills then you cannot escape the screen to a higher level, and on some of the screens your robot has to complete it in a special routine. Now if by chance you are a clever little R2D2, then the second part of this software package will interest you even more. Usually, if you finish a game, that's it — not so with Mr. Robot. With this game you can actually create your own levels and save them for a game later.

Using the joystick, you select various items such as moving walkways and bombs which, when you walk over them, they light, thus giving you a limited amount of time to get out of their way. Once a screen is finished, you have the opportunity to test it to make sure that it is feasible. Finally, as a footnote this package originally came from across the water and is therefore already tried and tested to destruction, which is what will inevitable happen to your robot!

S.L.F.P.

Mind Control
★ ★ ★
Mastertronic
£1.99
CBM 64

WATCH OUT ZYCO BECAUSE here come the scientists sworn to kill you Zyco is an almost indestructible alien who has taken over the earth and is busily enslaving us mere mortals. But he has one weakness. If you obliterate the nerve centre in his brain, he can be destroyed. You are the scientist who has been chosen to destroy Zyco, so you take a newly invented miniaturising pill and enter his brain. Once inside you have to work your way through the corridors of power avoiding the marauding white corpuscles and threatening anti-bodies, jumping and running to keep out of their way. But be quick because time is forever running out and you have to reach the nerve centre before the effect of the miniaturising pill runs out and you explode back to full size.

K.M.



Exodus
★ ★ ★
Firebird Software
£2.50
CBM 64 + Joystick or keyboard

CHEAP DOES NOT ALWAYS mean nasty: this game, at £2.50 is excellent value.

The idea is very simple. You move a spaceship in either direction around the edge of a pit from which all sorts of nasty objects are trying to escape, and you shoot as many of them as possible. Altogether there are eighteen different types of alien, ranging from rabbits to hover mowers. If you bump into one or run out of time you lose a life. This also occurs if you accidentally shoot "Spud" who is supposed to be a friend but is actually a menace! You also lose a life if you permit the escape of a mutant llama — why does that sound familiar?

If you manage to survive long enough, you move on from the first, octagonal pit to a harder, diamond-shaped one and eventually a square pit which is more difficult still. After that you return to the first pit but the nasties move faster.

In case that sounds too easy, your laser-gun is inclined to overheat, so your shots need to be rationed carefully and well-aimed.

There are some interesting sound-effects, and the graphics, though nothing special, are adequate. The whole game, once you get used to the unusual controls, is fast and quite compulsive. Well done, Firebird!

Alien
★ ★ ★ ★
Argus Software Press
£8.99
CBM 64

YOU'VE SEEN THE FILM, NOW YOU'RE A star in it as Captain of the ill-fated spaceship Nostromo. The film had a haunting atmosphere which the authors have succeeded in capturing in this masterful role-playing game, with suitably creepy sounds throughout.

The main screen display shows various deck plans of the three-tiered Nostromo. As captain you direct the other characters about their business, picking up weapons and equipment, even ordering them to rest when they are under too much stress. The seven other crew members have varying traits via the game's personality control system. Demand too much of them and they'll suffer a nervous breakdown. With the Alien attacking the crew, one of whom is a mysterious android, Jones the pet cat creating havoc with the tracking system and fires breaking out all over this space hulk, the authors forgot the poor old player who's likely to be a nervous wreck by the end of it all.

Alien is difficult but good fun if you enjoy getting to know people as exploring crew members' strengths and weaknesses is essential. The options for any single game are immense with many rooms, corridors and ducts to explore, various types of equipment to use and a host of special instructions. All moves are controlled from an on-screen menu and sub-menus using function keys for selection. Actions are either confirmed by crew locations being amended on the deck plans or by textual update with sounds. The novice will require a few attempts to become accustomed with the game but the authors have thoughtfully provided a short scenario option for the space-rookie.

The cassette is accompanied by a booklet giving clear instructions for play and a summary of the film. To appreciate the game's subtleties, it will help to have seen the movie which would also give the player some idea of what to expect. I hid under my cinema seat first time round — the game is true to the film. Only the bravest players should confront the Alien in the small hours of the morning. I'll say no more — the hairs on my neck are bristling again.

R.M.





Magic Carpet
★ ★ ★
Mastertronic
£1.99
CBM 64

YOU KNOW THESE MASTERtronic cheapies may stint on price but they don't stint on value for money. Magic Carpet itself is a fairly straightforward cave exploration game in concept but with a fairly high difficult level. Technique is the key. No matter how impassable the obstacles look, it is always achievable providing you have the strategy right. Poor old Aladdin is the chappie who has been chosen to retrieve the stolen treasure from the evil sultan. To do so he has to pass through innumerable caves riddled with deathtraps such as bouncing boulders, spears of death, moving floors and acid rain. Finally he has to fight his way past the spitting dragon to retrieve the treasure and, dare I say it, the magic lantern which will see him safely back home.

Flyer Fox
★ ★ ★
Tymac
£9.75(tape)£14.95(disc)
CBM 64 + joystick

DO I DETECT SHADES OF Korean airline disasters in this? Having emerged from training as the best of the bunch, you have been given charge of the world's most advanced 'defensive' weapon — Flyer Fox. Your mission in this piece of 3D arcade action is to escort the commercial Jumbo Jet through the once safe international skies and fight off the nasty MIG fighters which are intent on blasting it out of the sky. Using your radar you have to track them down and shoot them out of the sky, veering left and right to get them in your sights as you give chase. Fortunately you get an audible warning when they are close but it still doesn't make it all that easy when you are running out of fuel fast. On the whole this is not a bad game although the speed with which the enemy MIGs move around makes it virtually impossible to track them down on the radar. Oh yes and apparently it talks to you! Extremely garbled messages are supposed to give you the feeling of reality in your flight. The gimmick doesn't work but there again it doesn't detract from anything either.

K.M.

SOFTWARE SPOTLIGHT



Caesar's Travels
★ ★ ★ ★
Mirrorsoft
£7.95
CBM 64

ANYONE WHO BOUGHT **Caesar the Cat** will recognise the hero of this new offering from Mirrorsoft. This time he is part of the newspaper/software group's Early Learning series. This package is aimed at the three to nine year age group but can be used with younger children if they are helped by an adult.

Caesar's Travels is a book and a cassette set which is intended to help reading. The book contains what seems to be one story, but because of options on most pages which lead the reader down various different avenues of thought there are eighteen endings to the initial story. The book is delightfully illustrated and would make bedtime reading for several nights. It also has a colouring book section to add to it's appeal. The cassette follows the same stories as the book. It starts with the familiar scene of Caesar in the pantry, failing to catch the mice and breaking crockery in the pursuit of the rodents. He is caught by the scruff of the neck and thrown out of the house. His adventures now start. At intervals throughout the stories the child is given simple tasks involving counting, colour recognition and right left, up down recognition. This all adds to the educational content of the program. Choosing the options in the progrm has been made easy by the inclusion of an overlay which fits neatly over the function keys.

The progrm is beautifully illustrated and follows the high standard set in Caesar's Cat; Caesar really looks as if he is climbing a wall or jumping over a fence. Another aspect of this program is it's use of sound. You actually hear footsteps chasing him, you hear his pitiful mewing as you see him trying to swim out of a lake and you hear the screech of brakes and the thud as he tries to run across the road.

This is a well thought out package which my four year old loved to play and which will give hours of amusement.

M.W.

Time Traveller
★ ★
Audiogenic Ltd
£5.95
CBM 64

IF YOU'RE A BUDDING Doctor Who type, who thinks he might enjoy travelling about in time and space, then this latest text/graphic adventure from Audiogenic may appeal to you.

You're cast as a lone space traveller who is woken up (I don't know by whom) during a space journey from somewhere to somewhere else, and given the mission to save all time and space from the evil Graf Von Schwarzherzen, the baddy of the adventure.

By using 'latest technology' you must fully explore the craft and then teleport to and fro in time to recover...but then I'd be telling you the plot; suffice to say, it's another adventure where you have to collect certain things in order to save everybody from certain death.

The graphic part of the game is a very small window in the centre of the screen. The graphics are adequate while the sound is so minimal to be almost non-existent, except for an irritating little tune which plays every time you do something clever. The text panel is directly underneath and is qûite well done, being a teleprinter type simulation, complete with sprocket tracks. The only drawback to this set-up is that on more than one occasion I was able to type quicker than the program could, which meant several attempts at typing a command due to missed letters.

The standard verb-noun format is taken and all words can be shortened to four letters. This is quite useful.

Somebody really should tell the people at Audiogenic about fast loaders: the tape took over sixteen minutes to load. There was, also, no save game option.

All in all for the price of £5.95 its good value but don't expect the Hobbit, first time adventurers only please.

M.T.U.

Battle Through Time
★ ★ ★ ★
Anirog
£7.95
CBM 64 + joystick

FOR BATTLE READ BUGGY BECAUSE that's your sole source of protection and transportation. It's also the latest in time travel technology. The buggy is controlled by the joystick. It can be speeded up, slowed down, made to jump and fire bullets upwards and forwards. Starting off in the year 2525 (a great song that years ago if I remember rightly) you have to travel through the battles of time giving assistance in shooting down the various types of flying machines and creatures from bi-planes through helicopters and satellites to pterodactyls. On the ground you have to watch out for the pits in the terrain as well as blast the boulders in your way and score points off the enemy soldiers. To move from one period of time to another, you must complete a ten mile journey to reach the teleportation point within the set time period although a continuous game option allows you to pick up where you left off. Get off and do your bit for mankind in this highly frustrating arcade game.

K.M.

Zulu
★
Firebird Software
£2.50
CBM 64 + Joystick (or keyboard)

PERHAPS THIS GAME MIGHT have been called Pacman in the jungle except that it would be an insult to the famous Pacman!

Your task is to move around a maze, collecting — wait for it — gold tribal masks and pink spiders! You are being chased by ferocious Zulu warriors, contact with whom is instant death. Luckily you are protected by a force field which, if activated in time, will disolve Zulus on the spot. Here and there you find grey cooking pots, which do not turn you into soup of the day but instead make the maze invisible whenever you bump into one. Touching a second pot while in this condition loses you a life.

Fortunately there are exits from the maze, but if you reach one you just move into another, similar maze and start again. In all there are twenty-five such mazes, with exactly the same task in each one, so it should be a good game if you have trouble getting to sleep.

Graphically the game is fair, except that the mazes are quite small, occupying just over half the screen. The rest is taken up by score tables. There are some interesting sound effects too, but over all the game is monotonous, unoriginal and frankly silly. **PRB**



Cliff Hanger
★ ★ ★
New Generation Software
£7.95
CBM 64 Keyboard or joystick

NEW FROM THE PEOPLE WHO brought you Trashman comes Cliff Hanger, a computer game based not only on cowboys but also on the movies too.

When the game is first loaded up you are introduced to the movie star Cliff Hanger and his brother Coat. Cliff is going to star in a series of cowboy movies with you being the actor playing Cliff. Your job in all of these films is to stop the bandit, El Bandito, shooting up the canyon. You can do this by using a collection of rather strange methods and devices ranging from a one ton weight to a boomerang. Sounds intriguing? Read on.

Following the on-screen instructions you first have the option of using the keyboard or a joystick. The game uses a user definable key system which is a real plus in its favour. Also, I encountered a couple of problems with the joystick control and so it's safer to use the keys. Once you've made your choice you enter your name, the clapper board snaps shut and you're away.

The screens are generated randomly and so you could appear on any one of them: they include Circus Act, Chuckie Bomb, About as far as I can throw you, Boulder Dash, The See-Saw and The Cannon. At first this seems a very easy game. However, the truth could not be more different. I was playing it for ages before I finally got on to the next set of screens. This makes the game very infuriating and very soon it becomes repetitive and boring. This game is probably not the type that the average 'shoot-em-up' enthusiast would rush out and buy but it does contain a good humour element which will appeal to the younger compo-bashers as well as to the adult would-be cowboys. A good example of this kind of fun is when Cliff Hanger knock himself into the ground with a mallet.

Points are awarded upon the completion of each game at each level and with ten game situations on the first level this is no mean feat. The graphics in Cliff Hanger are reasonable and anything lacked by the graphics are certainly made up for by the soundtrack written by Brian Doe of Dave Dee, Dozey, Beaky, Mick and Titch fame. **R.B.O.**

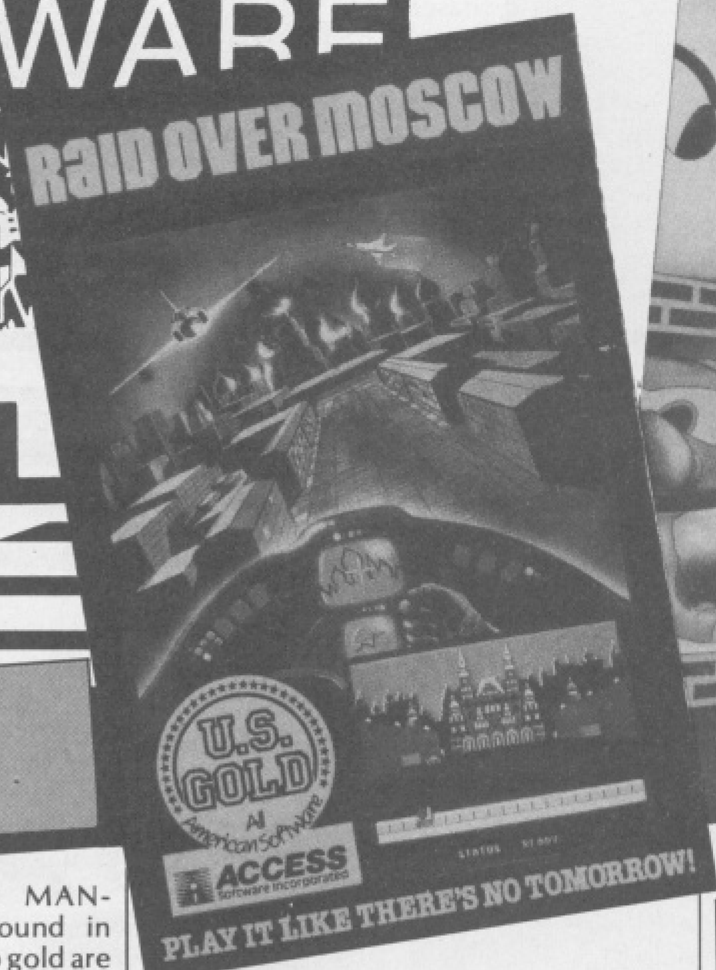
International Soccer
★ ★ ★ ★
Commodore
£14.99
CBM 64 + joystick

AT LAST AND WITH MANY THANKS TO Commodore's Kim Booth we can bring you a review of the company's very own international soccer package. What's more, the neat ROM cartridge which plugs into the expansion port in the back gives you instant access to the action.

The game is real match of the day stuff. It can be played either against the computer or against another player. If you want to play against the computer all you do is select the skill level on a rating of 1 to 9. Even for first timers skill level one is a bit of a push over, six is pretty even and nine is a real top of the table stuff. You can also choose the colour of the players' strip and, so you know which player is in control of the ball and which one of the opposition can do the tackling, they appear in slightly lighter shades. There are seven players on each side so they can vary considerably. Player movements are controlled only by the joystick. To pass the ball and shoot simply press the fire button. If you are defending, the goalkeeper automatically moves in the direction of the ball but the fire button initiates a dive. Free kicks, corners, throw-ins and goal kicks, they're all here in a game that gets more realistic and fun to play the more skillful you become. Just so that you know all is not fair in love and war, the player with the ball tends to be a bit of a slow coach except when he's heading it down field.

K.M.







Magic Stone
★
Audiogenic Ltd
£5.95
CBM 64

HAUNTED TRANSILVANIAN MANsions, magic stones, books bound in human skin and lead turning into gold are the things this graphic text adventure from Audiogenic is made of. The things you will most remember however are the sixteen plus minutes taken to load (yet more tea!), no save game option and no abbreviation of commands except the compulsory directions and inventory; you type everything in full, usually over and over again.

To be fair the adventure itself is very playable, with a small but adequate graphic idea in the middle of the screen and your commands and responses appearing undramatically underneath. The usual two word noun verb system has been used for your input with the program showing a reasonable amount of tolerance in its vocabulary.

You are set the task of collecting various items cunningly hidden by the evil Graf Von Schwarzherzen, the villain. When these are collected together in the right room of the mansion you will be able to transform lead into gold.

All you are given in the way of help is a magic amulet, reputedly from darkest Africa which changes colour in the prescence of black magic, you also get a rather too helpful help command.

The major criticism of this adventure is that it almost takes longer to load than it does to solve — its far too easy. It's fine if you're a novice adventurer but pure "canon fodder" to any adventure buff. I recommend it as a first adventure but stay clear otherwise.

M.T.U.

Raid Over Moscow
★ ★ ★ ★ ★
US Gold
£9.95 cassette/£12.95 disc
CBM 64 + Joystick

MESSAGE TO RONNIE: NUKING THE Ruskies isn't such an easy task, not if this sequel to the excellent Beach Head is anything to go by.

There are seven scenarios to master in order to reach a successful conclusion. The really good thing about it is that there is a demo facility which allows you to take control at any stage and practice your skills. So what do you have to do? First you have to get your planes out of the hangar. You can do this one at a time to attack the Russian missile site or you might decide to take several out and have some on standby. To reach the enemy silos you have to make an attack run through enemy airspace. Unless you fly very low, the radar will pick you up and then guess what, they shoot at you. You have to destroy all the missile silos before you can make your attack on the Soviet defence centre where you have to kill the protecting soldiers, destroy the tanks and open the doors to get inside the reactor room. Once inside you have to neutralise the reactor room robots to make the nasty little thing overheat. Escape alive and you win.

Should you choose to accept this mission, it's pretty hard but nevertheless pretty good — albeit in questionable taste.

K.M.

Jet Power Jack
★ ★ ★
Micropower
£7.95
CBM 64 + Joystick (or keyboard)

I DON'T USUALLY LIKE PLATFORM AND ladder games much, probably because I'm not very good at playing them, or vice versa. Because of this the program was double tested by an independent panel of dedicated players aged 9 to 14.

As a variation on most other games of this type the ladders are dispensed with and in their absence you are provided with a jet pack which is activated by pressing the fire button.

I found this preferable to the usual method of fire +stick to jump.

You are in a space garage, so the story goes, and must collect randomly placed fuel pods from one side, before returning to re-fuel a randomly placed space ship on the other. Just to make things interesting there are strange bouncing objects in the way (aren't there always?). You lose one of your three lives if you touch these or the edges of the platforms which are, of course 'live'. You also have a limited supply of oxygen, indicated by a gauge at the top of the screen.

With on screen scoring, five levels of increasing difficulty (you may start on any level), a high score table and excellent graphics and sound, the game is better than many others of it's kind.

The general consensus of opinion from the panel was: difficult but great fun.

D.J.T

Space Ace 2101

★ ★ ★ ★
Ozisoft
£4.99
CBM 64 + Joystick and keyboard

NOW I KNOW HOW HAN Solo felt when he was a mercenary before Star Wars! This is an intriguing game. Not only are you concerned with wiping out the aliens, but also with the maintenance of your ship and indeed keeping yourself out of galactic prison when fines are imposed on you and the cash has run out! The idea of this game is to destroy the alien robot factory which is stationed somewhere in the Mhiyken system of planets.

Once the game is loaded, you are given a certain amount of money with which you buy fuel. To gain more funds you have to leave the safety of the fuel station or the intergalactic hyper-market and tackle some of the nasties waiting for you outside. When you shoot an alien a bounty is paid and if you survive to enter the atmosphere the counsel of that planet will pay you. If you have committed an offence they will ask for payment themselves. Not only can you do the above but since there has been a little light colonisation you can also transport the occasional traveller to another planet. This will, again, enable you to raise the money for fuel. I think the best way to describe this game is as a monetory arcade adventure. The graphics on this program are not of exceptional quality but when you compare the size of the program something had to be cut down. The quality of the sound on this package is, again, lacking in that all important 'zip'. If all things are taken into consideration about this program it is quite good, but for me at least the graphics let the side down.

Eddie Kid Jump Challenge

★ ★ ★ ★
Memotech Software Communications Ltd
£7.95
CBM 64 + Joystick (or keyboard)

COMING COMPLETE WITH an Official Contender Card and an Eddie Kid sticker, the Eddie Kid Jump Challenge is a must for any budding BMX fanatics. Once fastloaded in, I saw an eye-soar — standard graphic blocks to display the title page. I must admit that at that point I was a little sceptical of the rest of the program, but when I entered the actual game it soon redeemed itself.

The basic idea with Jump Challenge is to jump as many cars as possible without crashing your motorbike. Sounds easy enough, but when you play it, it is a totally different story altogether. When attempting a jump you have to gain the correct speed and position on your bike. Failing to do that will lead to the inevitable crash in which we witness Eddie Kid bouncing along the ground! I must point out that the review copy did not contain the "jumping barrels on a bicycle" section so I shall just concentrate on the motorbike section. To control your motorbike you use either the keys or a joystick; changing gears has been put on the function keys and the brakes are on the space bar.

One point I found amusing during playing the game was that when you crashed, and after Eddie had hobbled off the screen, it displayed a message asking whether you enjoyed hospital food! I found that after crashing several times and being asked the above, I was dying for the screen to clear so that I could try again. You can actually have some fun popping wheelies and crashing (shows what sort of mind I have).

S.L.F.P.

Frenzy

★ ★ ★ ★
Micro Power
£7.95
CBM 64 + Joystick (or keyboard)

I FOUND THIS A DIFFICULT PROGRAM to review. Upon loading, my first impressions were of a game with only adequate sound (which, reminded me of a TV commercial for cigars, very soothing) and graphics which are, to be fair, less than adventurous. You may by now be wondering how come I gave it four stars. Well things are not always as they seem and after playing for half an hour or so it dawned on me that not only was I having fun but that the game is really quite complex.

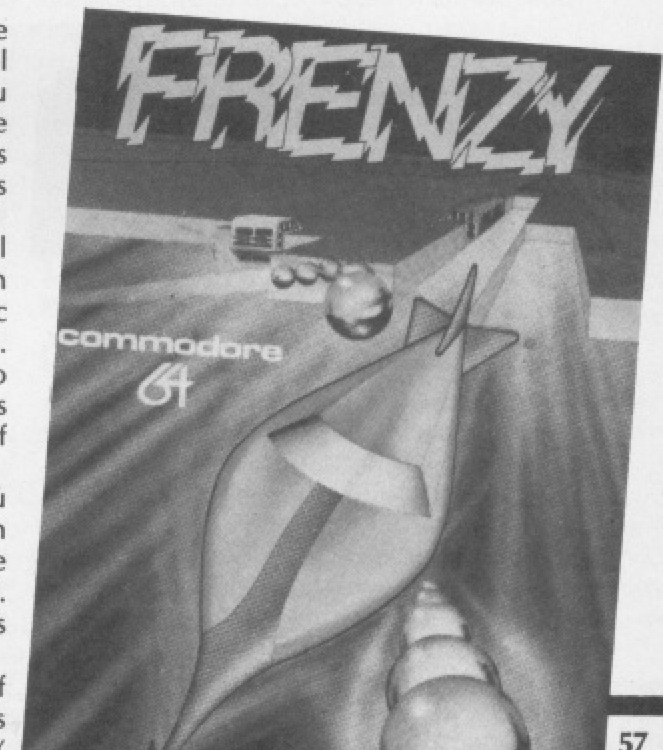
To set the scene: you pilot a robot craft around the edge of a scientific research centre within which roams a string of subatomic particles. By pressing the fire button your craft can be driven into the centre, leaving a green trail as it goes. By then driving to any side, the area enclosed by the trai will fill with colour, the object of the game being to trap the particles within the coloured area and so destroy them. Two things hamper your effors. The first is that if the particles touch your trail before you complete a move then you lose a life. Secondly, at higher levels there are small shining things called chasers travelling along your trail whose touch is fatal.

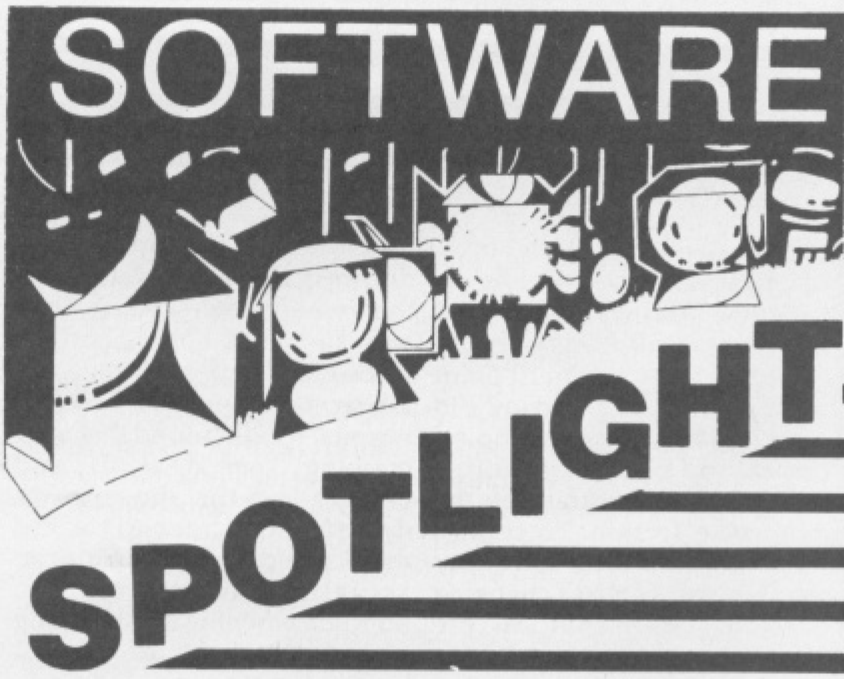
The 'robot craft' is in reality a small diamond shaped object, the 'research centre' a blank rectangle and the 'atomic particles' look like a string of small beads. However, I found the game to be so engrossing that the lack of fancy graphics did not detract from the pleasure of playing it.

Your score and remaining lives (you start with three and gain an extra one on completion of levels 3,6 and 9) are displayed along the top of the screen. Down the side is a gauge which indicates the percentage of screen filled.

In later screens the number of particles and chasers increase as does their speed. At first glance similar to STIX from SUPERSOFT, FRENZY for my money is the better game, simple in concept yet addictive and definitely value for money.

D.J.T





Mastercode Assembler

★ ★ ★ ★
Sunshine
£14.95
CBM 64

LEARNING ASSEMBLY LANGuage can be a daunting task, especially if you're learning from a book and assembling by hand. Try it and see! This program is a nice friendly utility which will be of value to beginners. Probably the most user unfriendly aspect of the program is the long time it takes to load. Judging from it's size and speed of operation, I guess that the program is written in BASIC and compiled. Not that I have anything against BASIC, but the length of the main program limits the amount of RAM available for source codes to 15K. More of that later.

The package is essentially an assembler, disassembler and simple monitor. Your source code is created with the aid of a file manipulation routine. The code is entered as numbered lines similar to those used in BASIC. The assembler uses two passes and therefore allows you to use labels for loops and memory locations. The usual pseudo op-codes for tables and specifying assembly location are supported. The source code can be saved and loaded from storage at any time. The assembler has the usual facilities allowing symbol tables, full listings and output of errors. Source code can be assembled to memory or to device. The latter is handy if the object code overwrites any memory currently in use. Overall the assembler functions well at a fair speed, but I haven't had a chance to see how it handles a source code of significant length.

To assist in debugging your object code, there are one or two handy tools. First you can disassemble any slab of RAM. The disassembler code is fully intelligible with all relative branches listed with their actual location. Probably the most useful part of the package is a trace facility which enables you to step through the execution of code without it ever crashing!!....what bliss. The only fly in the ointment is that the trace scrolls the screen mucking up any display your code may be generating.

The remainder of the package gives a simple monitor allowing the manipulation of object code. I'm rather surprised that a proper monitor is not included, especially since many of them are public domain.

On the whole this is a very handy package which is easy to use. The limitation of size and source could be a problem, especially since linked files are not supported. For this reason and the relatively slow execution (compared to machine code), I feel that this package isn't really suited to the serious programmer. For the beginner and enthusiast, it's unbeatable.

A.W.

Jet Boot Jack

★ ★
The English Software Company
£7.95
CBM 64

AFTER LOADING, THE TITLE SCREEN displays the player's options. The titles and options are large and chunky and hard to miss and give rather a cheap look to the whole thing. There is a tune accompanying the titles which sounds original but childish — somewhat like a hurdy-gurdy.

On the options page, you may choose 1 or 2 players and the skill level (being a practice level or one of 5 other levels, each harder than the previous one. The higher the skill level the greater the number of nasties to contend with and the fewer the number of fuel pods.

Once a screen has been completed, you may start your next game at any screen number up to that one, but not beyond. It is therefore possible to complete all the screens on the practice level and then shift directly to the last screen at any skill level required.

Jack the Lad zips across the screen by way of his super fuelled boots. When Jack is moved, sparks fly from his boots and his fuel consumption quickens. He is supposed to be whizzing around a record pressing plant collecting musical notes. However, each screen is basically the same as the one before but with extra lifts to move around on, resembling something from Manic Miner.

Jack can move up and down the screen by way of the lifts. If he stands over a lift gap, he is killed, but with the aid of his boots, he can overshoot these gaps without falling into them. Maybe the game could have been more exciting if he couldn't transverse the gaps at all. He can also travel on conveyor belts and conveyor trollies.

Collecting the musical notes increases your score. You may replenish your depleting fuel by head butting the overhanging vinyl pods containing boot fuel. Be careful however not to head butt the hanging rock formations. You can duck under these by pressing the joystick fire button.

At skill levels 1 to 5 you encounter the bugs and gremlins previously mentioned, hanging from the vault's roof. There are various types of bugs and gremlins but many can be disposed of by travelling above their heads and bouncing up and down on them.

I rather liked the graphic Jack, especially the way he shrugs his shoulders to duck under the rock formations. The remaining display did not impress me very much and showed very little imagination.

As this game is supposed to represent someone's trip around a record pressing plant, it would have been nice to have had some music of sorts in the background.

After a short while, the game became boring and very repetetive.

Possibly this game is geared to the younger end of the market and it should do quite well there. But I don't think this will be a best seller. Others may find it enthralling and addictive.

S.E.

Di's Baby

Bad Taste Software
£6.95
CBM 64

HAS THIS GAME HAD PUBLICITY OR has this game had publicity? Most press comments have concentrated on the abject bad taste of the subject matter. And me? Well I shall concentrate on the poor quality of the offering. The concept of the game is naturally royal and centres on whether or not dear old Charles and Di should be allowed to bring another baby into the world. The game is in five different parts, all of which fail to live up to expectation. The first screen is full of lavatorial humour with a lousy space invader imitation with Charles zapping incoming potties with toilet rolls avoiding flying nappies, etc. Screen two has Charles in a platform game trying to get past all the obstacles and get a bit of privacy with the lady. Enough said, I think. the only reason I've gone this far is because of all the publicity and I wouldn't want you wasting your money.

K.M.

Time Trucker

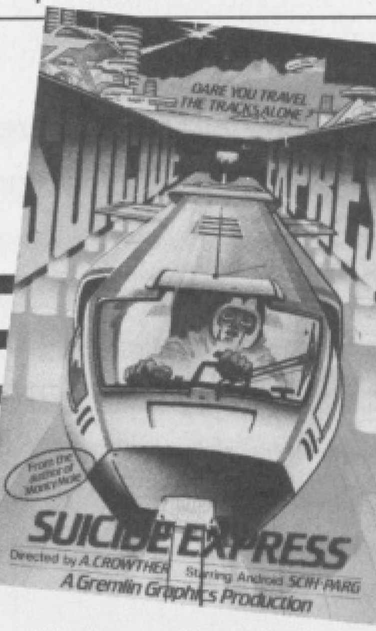
★ ★ ★ ★
ASK
£7.95
CBM 64 + Joystick

THIS IS AN EDUCATIONAL GAME WITH the main aim of teaching time skills. In it you take the role of a lorry driver going round farms collecting produce and delivering to a central depot. You have the choice to be one of three levels of truck driver and to go round one of three courses. Each choice of driving level gives a different game and the route maps give the difficulty factor; thus, there are nine possible games. The three routes are a map which is printed in the book, a secret map with random road blocks.

The three games all involve picking up the fruit and vegetables from various farms on the maps and watching the clock. In 'Trainee Trucker' you have to log your time at the various farms; this is done by converting the time on the analogue clock shown and putting it onto the digital display. Once this is done you can collect your order and deliver it to the depot. In 'Trucker' you must still collect the farm produce but instead of logging in you have to watch the time display as the farms open only at certain times and you are working in two hourly sessions. 'Truckers' is similar but the farms are open for only fifteen minutes each. Each of the games is great fun and, with the added difficulty levels, can be fast and furious.

The graphics and sound were of a high standard and 'Novaload' meant there was no long waiting for the game to load. Overall a good program which at it's higher levels benefitted from playing with a partner.





Suicide Express

★ ★ ★ ★ ★
Gremlin Graphics
£7.95
CBM 64 + Joystick or keyboard

WHAT A BIG YEAR IT'S BEEN FOR young Tony Crowther, author of such favourites as Blagger, Killer Watt, Loco and more recently from his new software house — Gremlin Graphics, Percy the Potty Pigeon and Monty Mole. Tony, is just one of the games authors improving the standard of British software with each release and his latest, Suicide Express, is no exception.

As we have come to expect by now from Tony, the graphics were excellent and a rollicking soundtrack by Sky played throughout the game also. The game itself is quite reminiscent of the arcade classic Super Locomotive and more especially one of his afore mentioned releases, Loco.

As soon as the game was loaded, a voice greeted me saying, "Welcome to Suicide Express", which acted not only as a bit of a surprise but also as a further excitement tonic for things to come. I had read from the inlay card that it was my job to drive the Suicide Express as android SCIH-PARG and to "clear the planet Nilmerg of all alien life and threatening dangers in order to make it fit for human habitation." So with that I pressed the fire button of my joystick and the voice said, "Get ready....Go!" I was off and straight away I could tell this game was going to be great.

The screen splits in two with a side on view of the train in the top half and a plan view of the tracks, your train and the nasties in the other. The super smooth 'wrap-around' scrolling graphics give a real feeling of speed as the Suicide Express hurtles down the tracks pursued by jet bomber aeroplanes and a hovercraft with other trains heading straight for it too. You have to be really quick on the fire button if you want to survive this game! However don't think you just blast everything in sight with a never ending stream of bullets; they have to be collected on your journey from ammo dumps at the side of the track. To add insult to injury don't think you can collect and hoard your ammo either because, if you collect more than 50 pieces of ammo, you explode anyway!

The driver of the Suicide Express certainly gets a raw deal but, as the old saying goes, "You can't have your cake and eat it." Other dangers include scavengers and watchers. It all seems very one sided but do not despair! Unlike Loco you can control the speed of the Suicide Express which, although maybe a seemingly minor point at first, can be a real bonus once you realise its potential in order to get the really high scores, a task made not too difficult as there are loads of points to be gained via the generous scoring system (you get 1000 points when you first start off as well as getting a further 1000 points each time you die).

After giving the game your best shot and finally losing your 3 lives, the voice reads out your score.

With 14 levels of play and 32 screens this game is certainly not for the faint hearted. However, the excellent scrolling graphics and the fabulous soundtrack make this game horribly addictive and yet another Crowther commodore classic. Dare you travel the tracks alone?

R.B.O.

The French Mistress (level A)
The German Master (level B)

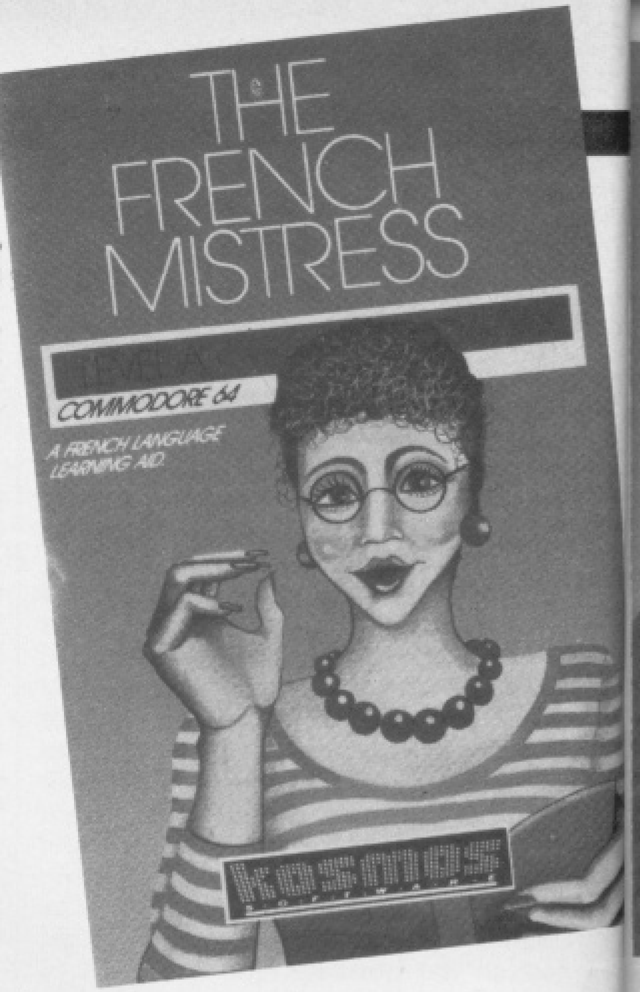
★ ★ ★ ★
Kosmos Software
£9.95 (each)
CBM 64

THESE TWO PROGRAMS HAVE THE same layout, method of use and aims so I have decided to review them as one. The two cassettes form a comprehensive language teaching program and have on each of them sixteen lessons and an overall control program. This control is recorded on side 1 of the cassette and, therefore, it is essential to load it before doing any of the seven segments on this side or the nine on side two. The control program contains the various options available; these include an option to load either a specific lesson or the next in the sequence and a variety of ways to run the lesson once chosen. You can type in your own words, phrases and translations to form new lessons; you can use the test mode to check whether you have learnt anything. You have a further option to save your lesson on a cassette once you have created it.

The languages are presented on two tapes each. Tape A contains lessons covering a wide range of every day vocabulary. You can learn the words for members of the family, parts of the body, shopping, the countryside, days, months, seasons, living creatures and food. Tape B has further vocabulary and has lessons on verbs and their tenses, adverbs, adjectives, conjunctions and other aspects of the grammar of the language being studied.

Both programs are well presented, easy to use and could be of use to anyone studying French or German either at elementary or exam level. It could even be of use to those contemplating foreign travel. I have found only one drawback and that is that there is no way to actually learn the pronunciation of the words in the programs. What a pity there is no sound tape to accompany them. However both of these sets could be of use as an adjunct to other lessons.

M.W.



The Magic Sword

★ ★ ★
Database Publications
£8.95
CBM 64

ADVENTURE GAMES WHICH CHILDREN can play are few and far between. The adventures on the market are generally too difficult for the average child because of the amount of reading or reasoning involved. This is a great pity because the strategy behind them is so important in the development of skills such as reading, logic and mapping techniques.

Now the software houses have seen the potential market and recently I have seen two: Creative Sparks' 'Dangermouse and the Black Forest Chateau' and 'The Magic Sword'.

The Magic Sword comes as a book and cassette package. The book is a delightful fairy tale telling the story of how Princess Poppy one day is so bored that she wanders around her castle home and its environs looking for something to do. Unfortunately she is captured and imprisoned by Bad Bertha the Witch, who then throws away the key. Her would-be rescuer Prince Freddy is turned, by the wicked witch, into a frog thus dashing all hopes — that is where the young reader comes in.

The cassette contains an adventure which anyone who has solved the Hobbit will find trivial but which should keep the under-ten occupied for a fair amount of time. There is no tricky keyboard entry to be undertaken. One letter commands have been built into the program and there is no need to 'look' as all that has been implemented. One criticism I have is that the text is all in upper case lettering (a mistake which adults tend to make when writing for children). However apart from this the package is delightful especially its simple but beautiful pictures in both the book and game.

M.W.

Ancipital

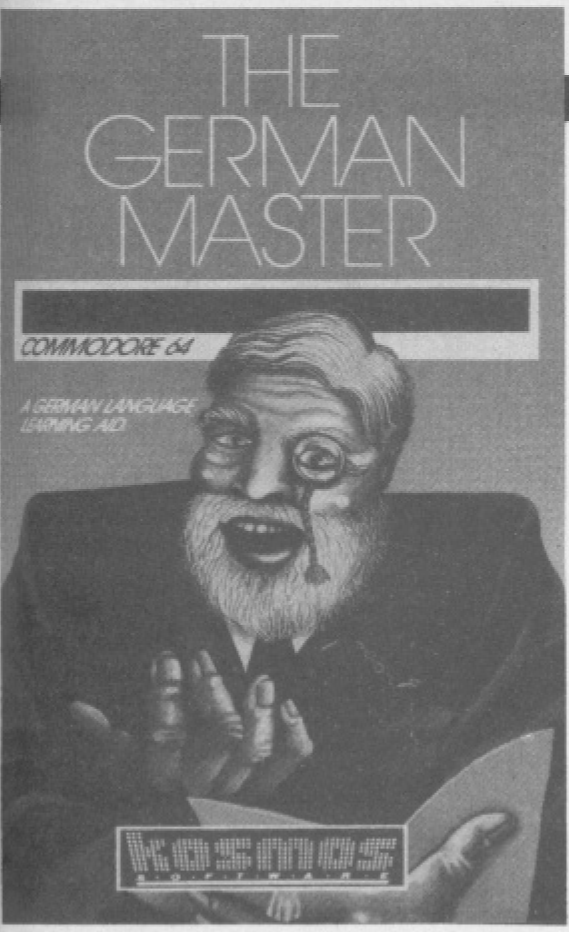
★ ★ ★ ★
Llamasoft
£7.50
CBM 64

ANCIPITAL IS JEFF MINTER'S LATEST contribution to the welfare of sheep, goats, Llamas and others. The game instructions are so comprehensively detailed that a user manual wouldn't have been out of place. If the instructions seem tedious, a bit of patience reading them will be worth it as the game would baffle anyone without them. Once play starts, all will become clear — well let's say less incomprehensible.

Although Mr. Minter takes a chop at adventure games in the forest of instructions, this game could, very loosely, be termed an arcade adventure. There are 100 screens or chambers to explore each offering different joystick fodder from alarm clocks to British Rail logos — everyone's pet hate is thoughtfully provided.

Ancipital, a goatish half-human jumps about from wall to wall — north, east, south and west — blasting the fiendish hallucinations until a sufficient number of them crash into a wall, weakening it, and allowing ancipital to enter the neighbouring room where, likely as not, another extermination technique is required. If any screen turns out to be too difficult, the player can always retrace his steps. But, beware being trapped in two or three adjoining rooms with a lack of baddies to zap. The whole package has the quality that has become the author's trademark — fine graphics too numerous to mention and background music and sounds maintaining interest from the intro track (courtesy of Genesis) during the fast Novaload until goaty's inevitable demise. If there's a spare gift voucher from Santa, it won't be wasted acquiring an Ancipital and lovingly caring for it — it shouldn't gather dust anyway.

R.M.



Henry's House
★ ★ ★ ★
The English Software Company
£8.95
CBM 64

AT FIRST GLANCE THIS APPEARED TO BE a game based on that old TV children's favourite, Hectors House. The screen display on the insert card seemed to point to this until I spotted the note "starring Little Henry", with a Royal crown above it. I then realised after reading further that this is a game depicting Little Henry's Royal Romp through the rooms of his new home. Whilst the packaging is unimpressive, I found the software enthralling.

The title display appears on screen to the tune of 'Rule Brittannia', after which you are treated to a brief demonstration of the 8 screens of the game.

You are Little Henry and have 3 lives to negotiate the game with. If however you lose all your lives, you may commence your next game from the screen you were last on instead of being returned to screen one. This is a superb feature and dare I say "Hooray for Henry".

Henry has come a long way since birth because for one so young he appears to be able to walk quite happily and is clothed extremely well.

Henry goes to screen 1, which is the clothes cupboard. Here he must collect gloves, boots, money bags etc and avoid the batty buttons and stomping boots. When all is collected (including the key to the exit door), Henry must make his way to the exit. Upon reaching this point, the screen clears and a display of Henry crossing a corridor from room to room is given which is quite delightful.

Room 2 is the bathroom, but Henry is not here just for the Royal Wee. Collect rubber ducks, soap brushes etc, avoid touching the sponge and bath taps and watch out especially for the dripping tap. If Henry pulls the plug at the top right of the screen, then the bath empties of water revealing further goods for our Royal Magpie to collect.

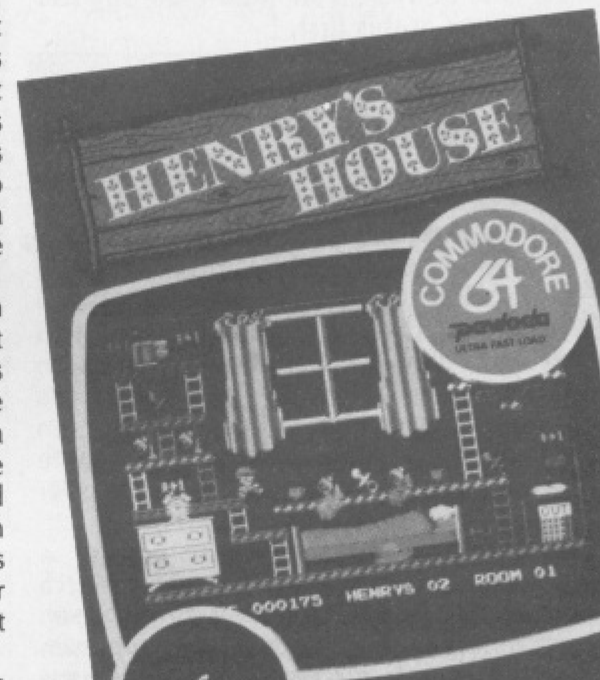
The kitchen can be a dangerous place for a youngster and this is so in screen 3. Don't get hit by the falling eggs, pop-up toast (with good sound affects) and falling tin can, but collect all the trifles, biscuits, cakes etc and avoid the hot tea pouring randomly from the teapot. Get to the door and exit to screen 4, the Lounge.

Again, objects must be collected, but watch out for the parrot escaping from his cage and flying around the room. Don't let him get you. I'll not give away any clues at this stage, but a sequence of collections must be set up. In the lounge is a cuckoo clock (complete with noisy cuckoo), a television set on the blink and a raging fire (where did they get their coal from).

And so to Henry's Playroom — screen 5. Each screen has different accompanying noises and this one is packed by a clockwork whirring. The floating 'Teddy' bears (sorry) a resemblance to Nookie Bear, but must be avoided at all costs. Collect the toys and parcels, jump into the aeroplane and then parachute down amongst the other toys (maybe Henry wants to be an SAS Soldier when he grows up), but don't get clobbered by Mr Jack-in-a-box.

Next is the nursery. Negotiate the obstacles for the collection of dummies, sums etc. Each item collected extends a ladder up to a shelf for you to collect items from that shelf.

Dinner is called on screen 7 in the dining room where goodies such as bananas, hamburgers and other eatables etc can be found. The 2 mad chefs can be nasty if you get in their way, so be careful. Negotiate the cooked turkey on the dining table.



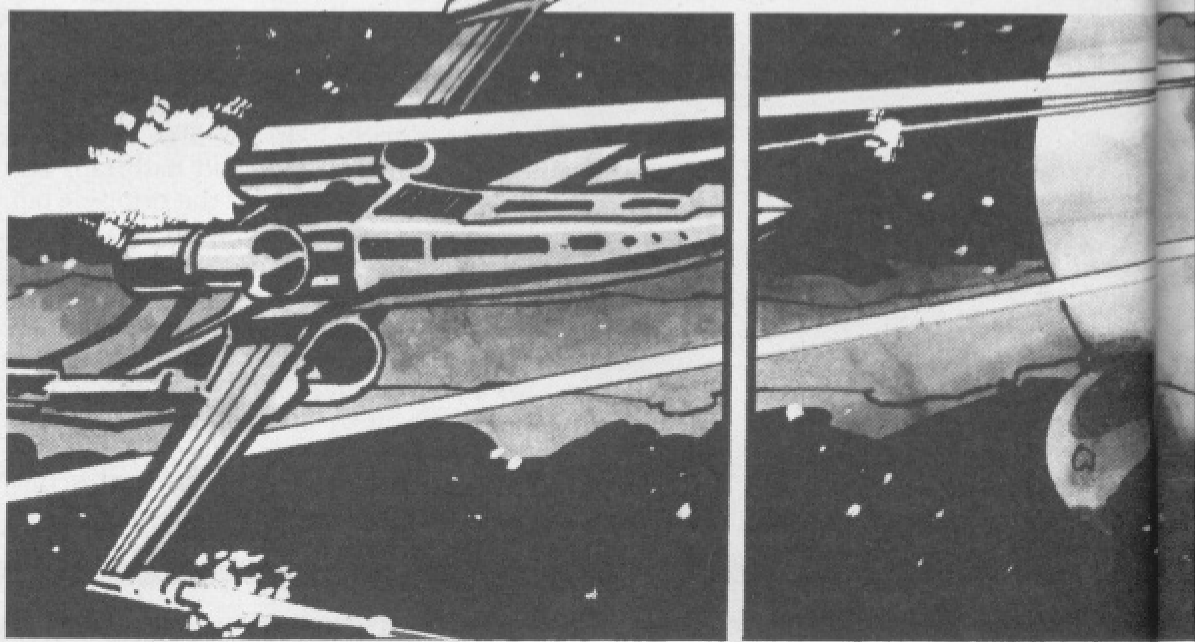
The hardest screen, of course, is last, Henry's Creepy Dungeon, complete with Witch (could it be an Aunty in disguise) and ghosts. The parrot appears again (in fact it could be a vampire), but this time it homes in on you (homing parrot?). Show him the sign of the crucifix to scare him off. Complete the round tour of the house and you end up back in the Clothes Cupboard at screen 1.

This game is very close to being a graphic adventure. Each screen is totally different from the others, which makes it a joy to play. There are a lot of surprises in store amongst the excellent graphic representation. In fact, it is close to being a cartoon of sorts. The link between screens is excellent. It will be a long time before I fall out with this game. I could play it time after time without becoming bored.

This game should appeal to all ages and I think prove to be excellent value for money. It is well thought out and different. **S.E.**

GAMES

Whether you want to write arcade or adventure games, you'll be interested in finding out from Pete Freebrey whether commercial games creators can help you.



MANY COMPUTER PROGRAMMERS are not entirely familiar with what BASIC will do, let alone machine code. They might very well have an idea for a new knockout game but lack the ability put it into practice. Even if the game could be coded in BASIC, it would be too slow to be worth considering.

There have been several games designers for the Sinclair Spectrum and the trend seems to be continuing for the Commodore machines.

A games creator should provide the user with a number of machine code routines that will first allow him to 'build' his game in simple, easy steps. It also provides him with an operating system to combine these various modules into a working game: the user simply describes what he wants and the program does the rest.

I have assessed five games designers — four for the CBM 64 and one for the VIC 20. One of these is an "odd-man-out' so I shall look at this first.

Scope 64 ISP
£17.95 (cassette)
CBM 64

Although **Scope 64** calls itself 'The Games Designer', it is not for those who are unfamiliar with writing their own programs. It does not actually help you *design* games but is an extension language that gives you 46 extra commands which greatly assist in producing a program that will handle graphics, sound and animation at a higher speed than ordinary BASIC.

Using these new commands, a program or subroutine is written in a series of REM statements. It is then compiled by the Scope master program and re-written into another specified area of memory. It may then be called via a SYS command (like a machine code routine — which it resembles) from your BASIC program.

In effect, it enables you to write machine code subroutines, but it does not help you design a program: you must already know how to build up a sprite or define the path your player will take when he jumps... If you know how, Scope 64 will make it easier and faster.

Complete programs could be written in Scope alone but I suspect most users would use it for handling subroutines that need a greater speed than BASIC can provide. Only single statement lines are permitted, so programs tend to get a bit lengthy quite quickly! The documentation is well sprinkled with examples and, at £17.95, it is well worth considering if your programming ability is ready for it. Once compiled, a Scope routine is independent of the master program and may be used in your programs.

Adventure Writer	**The Quill**
Codewriter	**Gilsoft**
£24.95 (disc)	**£14.95**
CBM 64	**CBM 64**

To all intents and purposes, these two programs are identical. Both are written by Graeme Yeandle. **The Quill** is available on both cassette and disc; **Adventure Writer** is on disc only. The latter appears to be a version produced for the American market now coming back to its country of origin! Some text is altered slightly but the core program remains the same — databases created on one will quite happily work with the other.

Adventure Writer comes complete with a working adventure ready to load or inspect via the operating program, although you will have to load Magician and not Sorcerer as stated in the instruction booklet!

These programs will help you produce a first class adventure program, with absolutely no knowledge of machine code or BASIC. All you need is the idea for the adventure itself, the locations, descriptions, objects to find and above all, the reasons for joining all these together into a linked adventure. The end product will run on its own, not needing the creator program to help it run.

I shall offer one word of warning though: adventures are, by nature, often complex in structure. Do make sure you have yours clearly set out on paper.

Of the two instruction books, Adventure Writer is the best. Both give you a worked example to key in but Adventure Writer gives a key by key account that is slightly more helpful.

On loading the program, you will be presented with the main menu. From here you can start to produce your masterpiece, commencing with your locations and their descriptions.

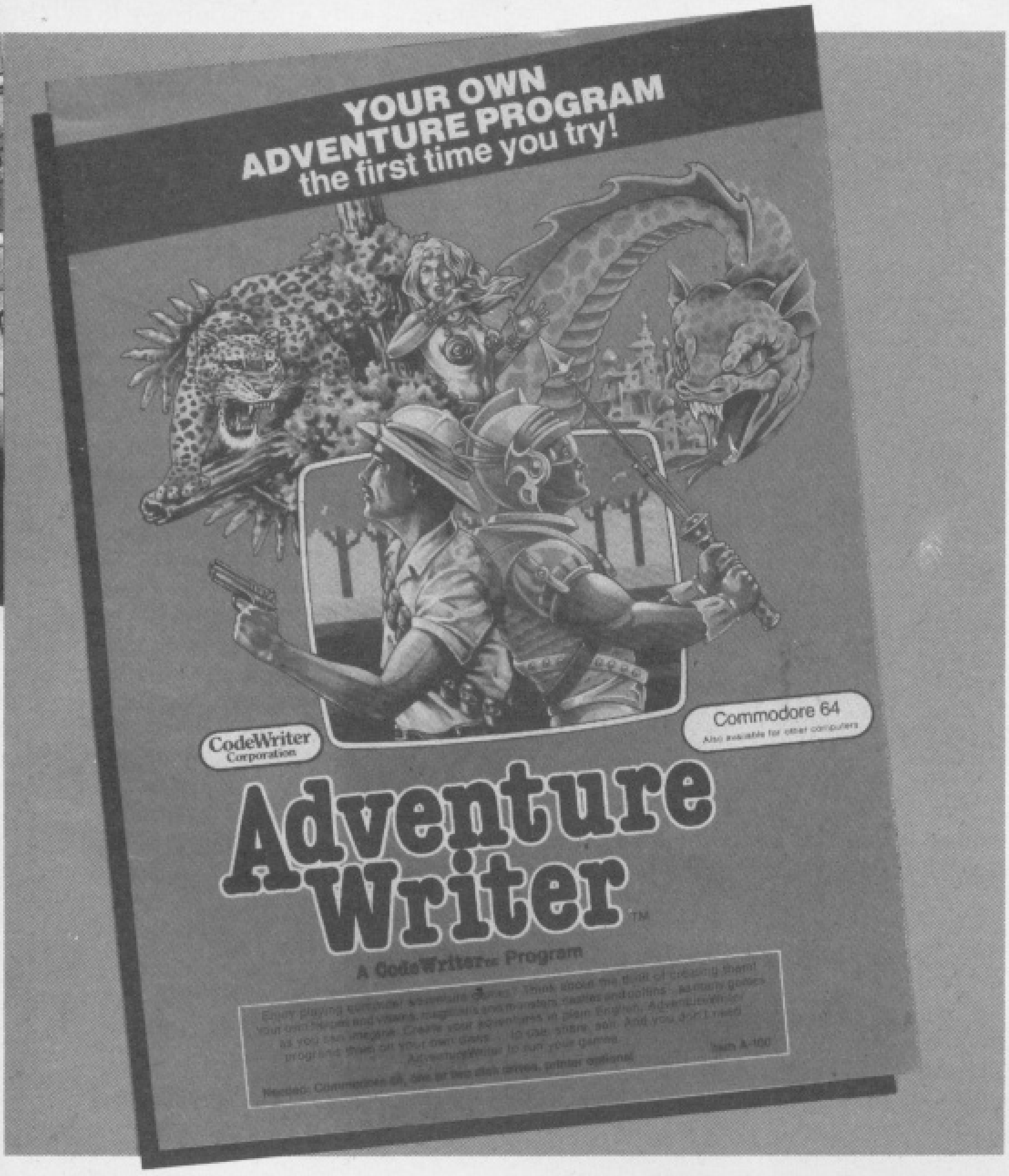
The programs use the 'business mode character set' (upper and lower case) plus the graphics symbols shown on the left of each key — accessible via the Commodore Shift key. Although you may be a bit daunted by the lack of the other graphics characters at first sight, it is possible to produce suitable pictures to accompany your text.

Once you've typed in your locations, you then move on to the Movement Table, Object Descriptions, Object Starting Locations, Messages, Vocabulary, etc. Each step is logical and well documented. You can even check how much memory you have left to play with.

While typing in text (graphics), all sixteen colours (and reverse) may be used to advantage — just remember that if you are using a monitor, an ordinary TV might not have the same colour separation/resolution. So, refer to the CBM instruction book for good colour 'mix and match'.

Complete games or databases may be

CREATORS



saved to disc or tape — save your growing database fairly often, just in case you encounter a problem. If using a disc drive and you get an I/O error, this may inhibit you when saving your database — to disc but not to tape (always have a spare cassette handy).

The only criticism I have of the progrm is that you cannot overwrite (on disc) a database with the same name. This means that, if you are writing a large program you may have disc management problems.

Both programs do what they set out to do and it is surprising what different types of 'adventure' may be created with the same basic tool. Neither company limits the use of any program you write, but they do ask that you mention their company name somewhere within the program. If the adventure is good enough, they will even sell it for you.

The Games Designer
Artificial Intelligence Products
VIC 20

The VIC 20 in it's unexpanded form (without extra RAM packs) has been sold in it's hundreds of thousands and sadly does not always get the attention it's users might wish. Artificial Intelligence Products (AIP) have produced a very clever games designer package that needs no extra money.

The cassette includes three games for you to play and/or edit. The games are simple but reasonably effective and in themselves will prove a fair amount of fun. Using the editing facilities it is possible to either build up a game from scratch or alter the existing games to your choice.

AIP claim that in using their methods of games creations, you will get the equivalent of 19K within the 3.5K VIC memory. This is achieved by loading the editing programs one at a time — performing one operation (or an associated series) and then storing the results by POKE the data to an isolated block of memory.

Having finished one aspect of editing, the next editing program is loaded — this does not disturb the *data* in memory but of course overwrites the previous *program* and effectively uses the same program area again and again...whilst building up a steadily increasing database for your game.

Having created your database you finally load the 'Main Game Base' in order to play the game. This program provides the operating system that uses your stored database.

Each game has the same basic pattern — you have to guide your 'player' from HOME to TARGET, avoiding various moving objects along the way...and then return. Each time you reach either location you score points and the speed of the moving objects increases!

Although this format may appear restrictive, it is surprising what variations you can build into such a simple scenario.

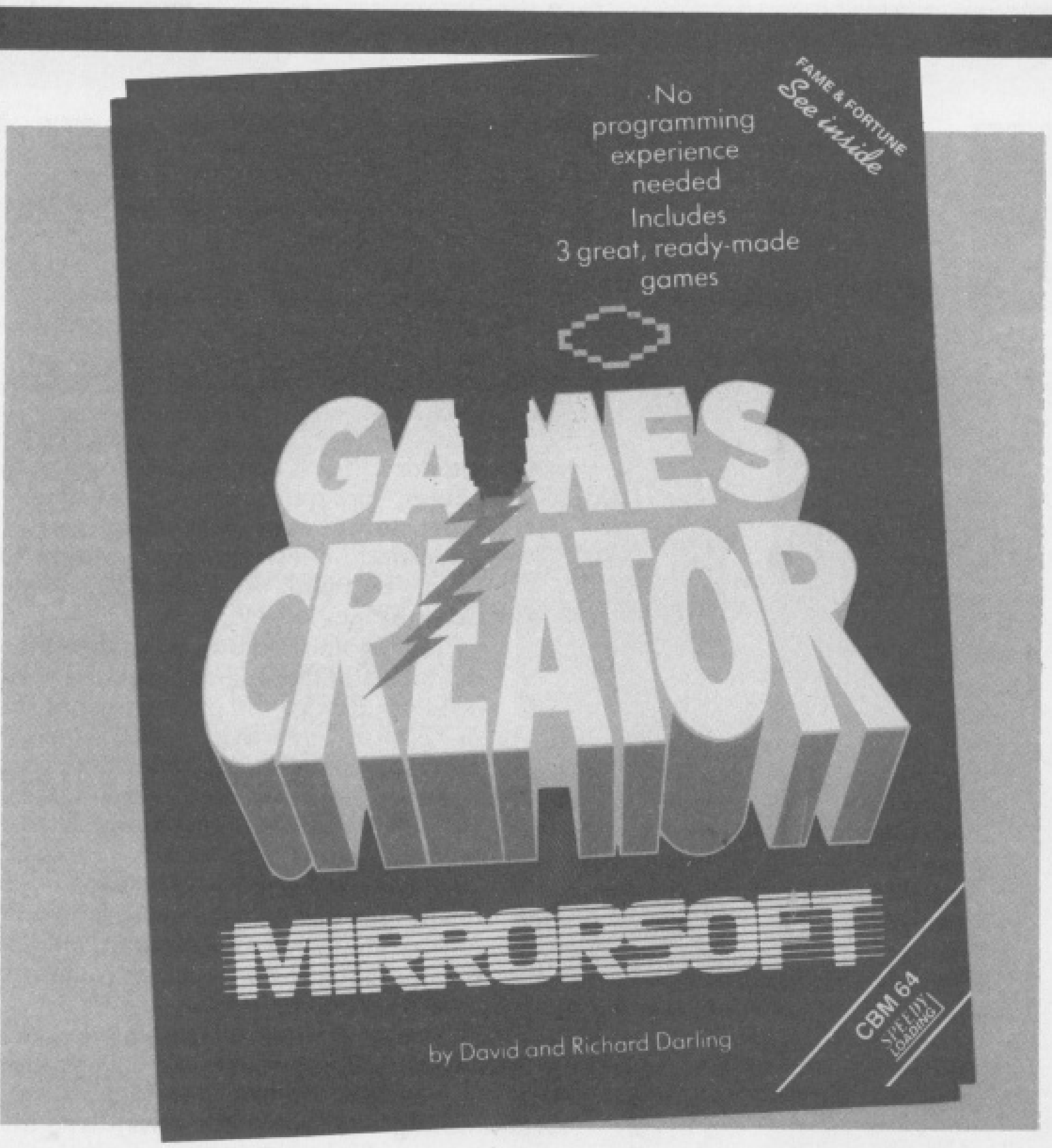
The editing sections consist of: Graphics Editor, Screen Editor, Tune Editor, Sound Generator, Game Formulator and Game Saver.

The Graphics Editor allows you to redesign 30 characters (each 8 x 8 pixels) from which you design your game. Four of these are used to create your 'player' — a different shape for the four possible directions of travel. The 'Aliens', opponents, call them what you will (!) are larger, being made up of a block of four characters; you may design two versions — left and right handed.

Two characters are specified for HOME and TARGET and the other 16 make up your scenery. A simple to use, expanded design panel is displayed as well as the 30 characters.

The Screen Editor enables you to design the screen display, including the position of the aliens, HOME and TARGET. Meanwhile a scrolling reminder of what each defined character looks like appears along the top of the screen. Each character placed on the screen may be one of eight colours.

The Tune Editor allows the creation of a theme tune of up to 50 notes covering two octaves (with sharps!). You may listen



to your tune at any time as you create it and it is easy to change any individual note. The Sound Generator provides three additional sound effects for collisions and reaching HOME and TARGET.

The Game Formulator defines the speed of the game, what keys will be used (if not using a joystick), points scored and also the contents of a scrolling message across the bottom of the playing screen.

Game Saver, does just that — saves your game database to tape for future playing or editing.

Overall, considering the memory limitations of the VIC, this game creator gives you several interesting possibilities to explore. Although the moving characters do not alter as they move, the scrolling is smooth and effective.

The games may only be played using the 'Main Game Base' so it is not possible to create a game independent of the 'Games Designer'. There can be no profit from this package but at least you can get a lot of fun from only 3500 odd bytes!

Games Creator
Mirrorsoft
£12.95 (cassette)
CBM 64

This program comes on either disc or cassette, the latter using one of the now common rapid load systems — loads in 3¼ mins. It provides the user with the 'master creator' (redesigner/editor) program that already has one example game in memory. This you may either play or edit; the databases for two further games are on the reverse side of the cassette.

The games supplied are on the 'Manic Miner', 'Scramble' and 'maze with nasties' types and provide reasonable games with good graphics.

The editing/creating facilities are very good, are menu driven and are generally easy to operate. Up to five animated 'aliens' are allowed, each with four stages of animation. The 'player' has eight stages — two in each of four directions. Also catered for are player/alien bullets and player/alien explosions.

All sprites are created in a block of four characters (16 x 16 pixels) and are multicoloured (four colours). The sprite editor is very easy to use and sprites are shown in both full size and also enlarged in the editing display panel.

The background scenery editor is very versatile. Three basic types of scenery are available: those that, 1) the 'player' may pass through 2) the 'player' may walk on or be stopped by 3) will kill the 'player' on contact. 49 different graphics blocks may be created for each of these three types, allowing a complex background to be built up relatively easily. The background can remain stationary or may be scrolled from right to left either slowly or quickly (for scramble type games).

The tune to be played throughout the game repeats itself but the main theme can be quite long and one of five instruments may be selected to play it. In choosing to write this theme tune you do *not* have the facility to view or correct a tune already written but *have* to start from scratch. Once writing the notes (no sharps) you may 'play' it at any time and if necessary delete notes from the visible 'page' — but no further back (seven 'pages' of music may be written). The music produced can be very good but you must get it right the first time...or start again from the beginning of the tune!

Sound effects for 'player' dying, shooting 'aliens' etc are catered for and may be editid. Again this is easy to implement but the variations possible are not all that great.

'Alien' movement may be very complex — up to 200 defined steps being allowed. Speed of animation, rules concerning collisions and how many aliens may exist are all variables you have at your control, allowing a very flexible and involved game to be developed.

There are a couple of weak links though... One is that you must load the master game creator first to play any game — it is not therefore possible to create an entirely independent program. The second and perhaps more serious weakness is that you can only create one 'screen', so that when you have completed/solved level 1, level 2 and up are exactly the same but with the option only of increasing the speed of the 'aliens'.

Conclusions

Scope 64 is rather a special case and should only be considered by those who already have a fair understanding of programming. It works well and a compiled routine is not dependent upon the master program. You may use programs written with Scope 64 as you wish; there is no restriction on sale of a compiled routine.

The two adventure writers The Quill and Adventure Writer both produce good quality products, indeed there are already a number of programs created using these programs, on the market.

The two arcade creators, Games Designer and Games Creator both require the master program to be used, to function — selling games, so produced, is not possible, except through the publishers of the programs themselves. Nevertheless, they enable you to put together interesting programs for your own use and also give you the opportunity to demonstrate your potential as a games designer — and software houses are always on the lookout for good ideas.

M.O.B. MAKER-64

Create your own Moveable Object Blocks (sprites) with John McHale's extremely powerful and versatile utility — an enhanced version of his Sprite Designer in issue 1.

Type in the listing and save it on a cassette or disc. Now RUN it and, if it has been entered correctly, you will be given the option of saving the object code to either tape or disc. On the other hand, if it hasn't, a screen error will appear. There are two possible error reports:

1) (Not enough/too many) data strings
2) Difference in checksum figures

Reports #1 speaks for itself. There is a total of 512 data strings to be entered. If you go above or below this figure, you will be presented with report #1.

If you receive report #1, you will almost certianly receive report #2 as well.

Locating the source of a checksum difference is much more difficult than tracing a 'DATA STRING ERROR', as you must check each 'Hex string' separately until the offending line has been found.

However, if you have entered the program correctly, you will now be given the option of saving the program code to either tape or disc.

It will be saved under the filename 'MM64'. When it has been saved, verify it to ensure that there are no errors present.

You are now the proud owner of an extremely powerful and versatile sprite designing utility.

Using M.O.B. Maker-64

Tape users may load the code by typing any one of the following statements.

1) 'LOAD'
2) 'LOAD"",1,1'
3) 'LOAD"MM64"'
4) 'LOAD"MM64",1,1'

It does not matter which of these you use, as the machine code will automatically relocate to the address from which it was saved — i.e. $C000 - $CEFF or 49152-52991.

Disc users can load the code by typing either 'LOAD "MM64",8' or 'LOAD "MM64",8,1'.

After loading is complete (approximately 2 minutes for the cassette version), you should type 'SYS (64738)' + 'RET'.

This restores the '64' to power up condition, resetting any system pointers/vectors that may have been corrupted by the LOAD.

Type 'SYS (52923)' + 'RET' to initialise 'M.O.B. MAKER-64'; the program title message, etc., will appear.

I have supplied a comprehensive guide to the various functions available to you and detailed explanations of their uses.

To give you some idea of the program's ease of use and versatility, I designed a complete set of 40 'PACMAN' sprites in approximately 20 minutes.

Function List

Function Number	Function Name	Keypress
1.	Cursor Home	'CLR/HOME'
2.	Clear Home	'CLR/HOME'+ SHIFT
3.	Cursor Right	'←csr→'+SHIFT
4.	Cursor Left	'←crsr→'+SHIFT
5.	Cursor Down	'↑crsr↓'
6.	Cursor Up	'↑crsr↓'+SHIFT
7.	Cursor Slow	' ; '
8.	Cursor Fast	' : '
9.	Enable Wraparound	'W'
10.	Disable Wraparound	'W'
11.	Enable M-Colour	'M'
12.	Disable M-Colour	'M'
13.	H. Fill	' . '
14.	V. Fill	' . ' +SHIFT
15.	H.Rub	'Del'
16.	V. Rub	'Del'+SHIFT
17.	H.Space	'Spc'
18.	V.Space	'Spc'+SHIFT
19.	Sprite Page +	'+'
20.	Sprite Page –	'–'
21.	Change MC #1	'1'
22.	Change MC #2	'2'
23.	Change Cursor Colour	'='
24.	Change Screen Colour	'RESTORE'
25.	Shift Sprite Right	'F7'
26.	Shift Sprite Left	'F5'
27.	Shift Sprite Down	'F3'
28.	Shift Sprite Up	'F1'
29.	Recall Sprite	'£'
30.	H. Line	'F'
31.	V. Line	'F' +SHIFT
32.	H.Wipe	'D'
33.	V. Wipe	'D' +SHIFT
34.	Reverse Video	'9'
35.	Invert Sprite	'I'
36.	(Rotate H/R 180° Sprite	'R'
37.	(Rotate M/C	'R'
38.	Rotate Sprite 90°	'R' +SHIFT
39.	Change Sprite Colour	'#'
40.	Transfer Sprite	'@'
41.	Swap Sprites	'*'
42.	Merge Sprites	'↑'
43.	Load Sprites (Tape)	'L'
44.	Load Sprites (Disc)	'L'
45.	Save Sprites (Tape)	'S'
46.	Save Sprites (Disc)	'S'
47.	Quit	'Q'

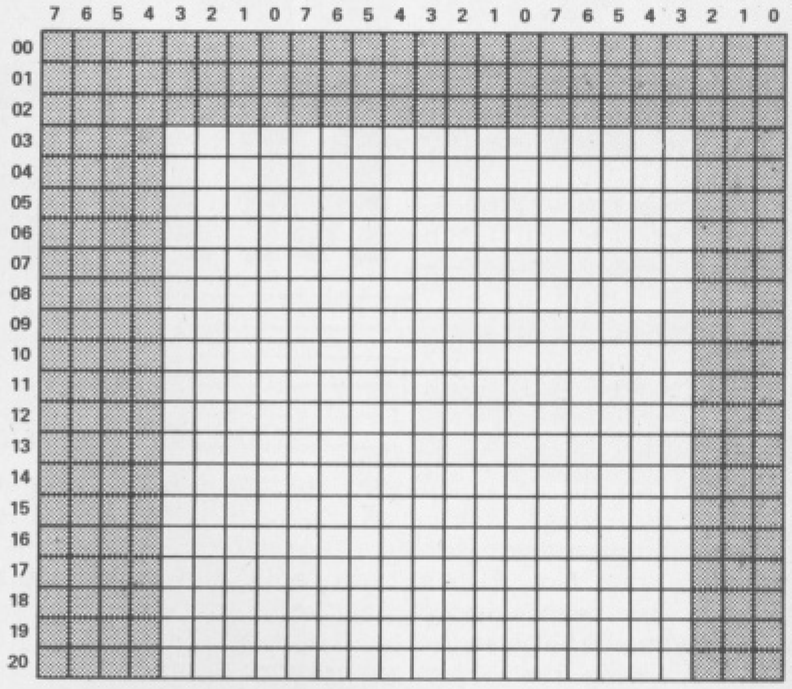
Explaining function uses

1. Places 'Cursor' at top left hand corner of grid.
2. AS ABOVE but also clears grid and sprite definition.
3. Moves 'Cursor' one place to the right.
4. Moves 'Cursor' one place to the left.
5. Moves 'Cursor' down one place.
6. Moves 'Cursor' up one place.
7. Slows cursor movement.
8. Speeds cursor movement.
9. When Wraparound is enabled, the cursor will re-appear on the left if it moves off the right.
10. When Wraparound is disabled, the cursor will stop when any of the four extremes of the grid are reached.
11. Enables Sprite Multi-Colour Mode.
12. Disables Sprite Multi-Colour Mode.
13. Fill one bit and moves cursor one place to right.
14. Fill one bit and moves cursor down one place.
15. Rubs out one bit and moves cursor one place to left.
16. Rubs out one bit and moves cursor up one place.
17. Rubs out one bit and moves cursor one place to right.
18. Rubs out one bit and moves cursor down one place.
19. Advances to next sprite definition.
20. Moves back to previous sprite definition.
21. Updates Sprite Multi-Colour register 1.
22. Updates Sprite Multi-Colour register 2.
23. Changes cursor colour.
24. Changes screen colour.
25. Moves Sprite one bit to the right.
26. Moves Sprite one bit to the left.
27. Moves Sprite one bit down.
28. Moves Sprite one bit up.
29. If you destroy a sprite definition by accident, this function will restore the sprite to its original condition.
30. Fills in all the bits from the left to the right of the grid.
31. Fills in all the bits from the top to the bottom of the grid.
32. Wipes out all the bits from the left to the right of the grid.
33. Wipes out all the bits from the top to the bottom of the grid.
34. Reverses Sprite Video.
35. Rotates Sprite through 180 degrees through the horizontal (X) axis.
36. Rotates a standard hi resolution sprite through 180 degrees through the vertical (Y) axis.
37. As above but with a multi-colour sprite instead.

Note: When 'R' is pressed, Function 36 is executed if the multi-colour mode is off and Function 37 is executed if the multi-colour mode is on.

38. Rotates sprite 90 degrees. (See Special Notes).
39. Updates Sprite colour.
40. Transfers a sprite definition from page x to page y, where x = the source page and y = the destination page.
41. Swaps Sprite definition x with sprite definition y, where x = the source sprite and y = the destination sprite.
42. Combines the definition of sprite x with the definition of sprite y, leaving the result in sprite page y, where x = the source sprite and y = the destination sprite.
43. Loads Previously Saved Sprite Data from Tape.
44. As Above, but loads from Disk.
45. Saves the desired sprite blocks to Tape.
46. As Above, but saves to Disk.
47. Exits to Basic.

Special Notes on Function Number 38



As most of you are probably aware, the CBM 64 'Sprite' is not perfectly SYMETRICAL ie. it is 24 bits wide by 21 bits deep and therefore it is not possible to rotate it perfectly through 90 degrees. Therefore you should avoid using the areas that I have shaded, when you are using this function, in order to get the best results. In any case, if you decide to use the whole sprite area, you may relocate the sprite definition using functions 25-28 inclusive and then fill in any bits that have been lost off the edges.

Special notes on functions 40 41 & 42

The same general rules of use apply to each of the three functions listed above.

As you will have seen in the notes on function uses, x = the source (start) sprite page and y = the destination (finish) block.

Before selecting any of these functions, you must select the source block by using functions 19 & 20; ie. 'Sprite Page+' and 'Sprite Page–'.

Now press the key associated with whichever function you use to use; ie. '@', '*' or '↑'. You will now see the sprite block number flashing at the bottom of the screen. Use functions 19 & 20 once more to select the destination block and once you have selected it, press 'RETURN' to execute the function. There are two important points to note which are as follows;

1. Pressing 'RUN/STOP' while engaged in any of these three functions will return the computer to normal code of operation without any changes having being made.

2. After pressing 'RETURN', the function will be executed and the sprite page number will be reset to that of the source block; ie. if you transfer sprite no. 137 to block no. 232, after accomplishing the task the page number will return to 137.

Special notes on Function number 45/46

When this function is called, you shall be presented with the message 'Save Sprite Data to Device'.

Below this, you will be prompted by 'From Block Number ?' and a flashing cursor.

Type in a number between 128 and 255 and press 'RETURN'. Now you will be prompted by 'To Block Number ?' and a flashing cursor once more.

Again, type in a number between 128 and 255 and press 'RETURN'.

You now are requested to enter the filename but remember, only the first 16 characters will be accepted.

Then you will be given the option of saving to tape or disc or abandoning the routine.

As a final point, I mention that any number out of the range (128-255) incl. should not be accepted e.g. 127 or 256.

But I have a confession to make! Here lies the only 'BUG' I could find in the entire program.

If you type any single digit number except (2) you will be presented with the message 'Input Invalid, Redo from Start'. You may well ask yourself why this is so and it took me quite a considerable time to figure out why myself.

The reason is that '2' is accepted as 200 and this number is within the 'Legal' limits. Likewise, '13' will be accepted as 130; '25' as 250 etc.

So the bug is not so bad after all as it may be used to your advantage.

Program listing

```
5 GOSUB10000
10 REM ::::::::::::::::::::::::::::::::
15 REM ::                            ::
20 REM :: M. O. B.  M A K E R - 64. ::
25 REM ::                            ::
30 REM :: DESIGNED & WRITTEN  FOR    ::
35 REM ::                            ::
40 REM ::'Y O U R C O M M O D O R E'::
45 REM ::                            ::
50 REM ::BY  JOHN MC HALE , SLIGO , ::
55 REM ::                            ::
60 REM : EIRE -- (C) NOVEMBER 1984. ::
65 REM ::                            ::
70 REM ::::::::::::::::::::::::::::::::
75 REM ********************************
80 REM *                              *
85 REM * CONVERT HEX STRINGS TO M.CODE*
90 REM *                              *
35 REM ********************************
100 SA=49152:CS=0:SC=0
110 READHEX$:IFHEX$="END"THEN200
120 FORC=0TO7
130 H=ASC(MID$(HEX$,C*2+1,1))-48:IFH>9THENH=H-7
140 L=ASC(MID$(HEX$,C*2+2,1))-48:IFL>9THENL=L-7
150 BYT=H*16+L:CS=0S+BYT:POKESA+C,BYT
160 NEXTC
170 SC=SC+1:SA=SA+8:GOTO110
200 REM *********************
210 REM *                   *
220 REM *  ERROR TRAPPING .  *
230 REM *                   *
240 REM *********************
250 IFCS=511650ANDSC=512THEN400:REM NO ERRORS/PROCEED TO NEXT MODULE
260 PRINT"■■■";:REM [CLR/HOME]/[WHT]/[RVS/ON]
270 PRINT"ERRORS MADE IN PROGRAM ENTRY AS FOLLOWS:■":REM [RVS/OFF]
280 IFSC=512THEN340
290 IFSC>512THEN310
300 PRINT"NOT ENOUGH ";:GOTO320
310 PRINT"TOO MANY   ";
320 PRINT"DATA STRINGS ENTERED":PRINT
330 IFCS=511650THEN360
340 PRINT"ERROR IN CHECKSUM COUNTER = ";ABS(511650-CS):PRINT
350 PRINT"YOUR FIGURE =";CS;" MY FIGURE = 511650"
360 PRINT:PRINT"CHECK PROGRAM LISTING CAREFULLY !":STOP
400 REM *****************************
410 REM *                           *
420 REM * HEX DATA ENTERED CORRECTLY *
430 REM *                           *
440 REM *****************************
445 POKE53281,6:POKE53280,14
450 PRINT"■■";:REM [CLR/HOME]/[BLU]
460 PRINT"HEX TO MACHINE CODE CONVERSION COMPLETE."
470 PRINT"THE OBJECT CODE OF THIS PROGRAM WILL NOW"
480 PRINT"BE SAVED TO EITHER  TAPE  OR  DISK UNDER"
490 PRINT"THE NAME ";CHR$(34);"MM64";CHR$(34);"."
500 PRINT:PRINT"■NOTE TO DISK USERS":REM [WHT]
510 PRINT"■------------------":REM [BLK]
520 PRINT"■THE DISK USED TO STORE THIS PROGRAM MUST"
530 PRINT"HAVE BEEN PREVIOUSLY FORMATTED AS THERE "
540 PRINT"IS NO FORMATTING ROUTINE BUILT IN , THE "
550 PRINT"DISKS USED FOR STORAGE OF SPRITE DATA   "
560 PRINT"SHOULD ALSO BE FORMATTED."
570 PRINT:PRINT:PRINT"PRESS ▮'■RETURN▮'■ TO CALL THE SAVE ROUTINE.";
580 POKE197,64
590 WAIT197,1
600 SYS52992
610 PRINT:PRINT"NOW  VERIFY  THE  CODE TO ENSURE THAT IT"
620 PRINT"HAS BEEN SAVED CORRECTLY"
700 END
900 REM ********************************
910 REM *                              *
920 REM * MACHINE CODE FOR M.O.B MAKER*
930 REM *                              *
940 REM ********************************
1000 REM :::::::::::::::::::::::::::::::
1001 REM :                             :
1002 REM : BLOCK 0  ( $C000 - $C0FF ) :
1003 REM :                             :
1004 REM :::::::::::::::::::::::::::::::
1005 DATA "8E8F328D818C98AD"
1010 DATA "A085989099ADA085"
1020 DATA "98909899A0859890"
1030 DATA "8F868E8F8EA09390"
1040 DATA "92699485A0828C8F"
1050 DATA "838BA08E958D8285"
1060 DATA "32A0BDA0A0A0A0A0"
1070 DATA "A09732819081928F"
1080 DATA "958E84ADA0A0A0A0"
1090 DATA "BAA08DAF838F8CAE"
1100 DATA "ADA0A0A0A99320D2"
1110 DATA "FFA98E20D2FFA908"
1120 DATA "20D2FFA90E8D20D0"
1130 DATA "49088D21D0A206BD"
1140 DATA "FFBF9D1A04BD05C0"
1150 DATA "9DE204BD0BC09DAA"
1160 DATA "05BD11C09DEA06A9"
1170 DATA "019D1AD89DE2D89D"
1180 DATA "AAD99DEADACAD0D7"
1190 DATA "A21BA9A09D6F07BD"
1200 DATA "1DC09D9707BD38C0"
1210 DATA "9DBF07A9019DFFD7"
1220 DATA "9D6FDBA90A9D97DB"
1230 DATA "A9079DBFDBCAD0DA"
1240 DATA "A2B7A0008A990204"
1250 DATA "990A0499120A490D"
1250 DATA "990AD8A90E9912D8"
1270 DATA "C8CAE0AFD0E6A900"
1280 DATA "85FB85FDA90485FC"
1290 DATA "A9D885FEA200A000"
1300 DATA "A9A091FBC891FB49"
1310 DATA "01#1FD8891FDA01A"
1311 REM :::::::::::::::::::::::::::::::
1312 REM :                             :
1313 REM : BLOCK 1  ( $C100 - $C1FF ) :
1314 REM :                             :
1315 REM :::::::::::::::::::::::::::::::
1320 DATA "91FD490191FBA5FB"
1330 DATA "18692885FB85FDA5"
1340 DATA "FC690085FC690485"
1350 DATA "FEE8E017D0D0A9B0"
1360 DATA "85FD85FEA92885FB"
1370 DATA "A90485FCA200A000"
1380 DATA "A5FD91FBC8A5FE91"
1390 DATA "FBA5FB18692885FB"
1400 DATA "A5FC690085FCA5FE"
1410 DATA "186901C9BAD004E6"
1420 DATA "FDA9B085FEE8E015"
1430 DATA "D0D460A90085FEA2"
1440 DATA "080A26FE06FC9007"
1450 DATA "1865FB9002E6FECA"
1460 DATA "D0EF85FD60AD04CF"
1470 DATA "85FBA94085FC205B"
1480 DATA "C1A5FD8D05CFA5FE"
1490 DATA "8D06CF602075C1A9"
1500 DATA "2A85F385FBA90485"
1510 DATA "FAA9D885FCA90085"
1520 DATA "02A000A2088B1FD0A"
1530 DATA "48300EA95185F7A9"
1540 DATA "0185F820CBC14CC1"
1550 DATA "C1A92E85F7A900F0"
1560 DATA "F068CAD0E2C8C03F"
```

Program listing

```
1570 DATA "D0D9608A489848A4"
1580 DATA "02A5F791F9A5F891"
1590 DATA "FBC8C018D015A5F9"
1600 DATA "18692885F985FBA5"
1610 DATA "FA690085FA69D485"
1620 DATA "FCA00084026BA868"
1630 DATA "AA60AD00CF85FAAD"
1631 REM ::::::::::::::::::::::::::::::::
1632 REM :                              :
1633 REM : BLOCK 2  ( $C200 - $C2FF ) :
1634 REM :                              :
1635 REM ::::::::::::::::::::::::::::::::
1640 DATA "01CF85FBAC03CF60"
1650 DATA "AD05CF85FCAD06CF"
1660 DATA "85FDAC07CF6020FA"
1670 DATA "C1B1FA297F91FA48"
1680 DATA "A5FB18690485FB68"
1690 DATA "C92ED004A900F002"
1700 DATA "A30191FA602016C2"
1710 DATA "AC03CFC8C019D01A"
1720 DATA "AD0FCFD00160A980"
1730 DATA "8D14CFAD07CF38E9"
1740 DATA "028D07CFA9018D03"
1750 DATA "CF608C03CFAD14CF"
1760 DATA "4AD005EE07CFA980"
1770 DATA "8D14CF60AD8D0229"
1780 DATA "01F0C22016C2AC03"
1790 DATA "CF88D01AAD0FCFD0"
1800 DATA "0160A9018D14CFAD"
1810 DATA "07CF1869028D07CF"
1820 DATA "A9188D03CF608C03"
1830 DATA "CFAD14CF0AD005CE"
1840 DATA "07CFA9018D14CF60"
1850 DATA "2016C2AE02CFE8E0"
1860 DATA "16D01FAD0FCFD001"
1870 DATA "60A9298D00CFA904"
1880 DATA "8D01CFA9018D02CF"
1890 DATA "AD07CF38E93C8D07"
1900 DATA "CF608E02CFAD00CF"
1910 DATA "1869288D00CFAD01"
1920 DATA "CF69008D01CFAD07"
1930 DATA "CF1869038D07CF60"
1940 DATA "AD8D022901F0B120"
1950 DATA "16C2AE02CFCAD01F"
1951 REM ::::::::::::::::::::::::::::::::
1952 REM :                              :
1953 REM : BLOCK 3  ( $C300 - $C3FF ) :
1954 REM :                              :
1955 REM ::::::::::::::::::::::::::::::::
1960 DATA "AD0FCFD00160A949"
1970 DATA "8D00CFA9078D01CF"
1980 DATA "A9158D02CFAD07CF"
1990 DATA "18693C8D07CF608E"
2000 DATA "02CFAD00CF38E928"
2010 DATA "8D00CFAD01CFE900"
2020 DATA "8D01CFAD07CF38E9"
2030 DATA "038D07CF6020FAC1"
2040 DATA "B1FA498091FAA5FB"
2050 DATA "18690485FBAD09CF"
2060 DATA "91FA60A203BD0ACF"
2070 DATA "09809DAD07AD0FCF"
2080 DATA "D005BD17C0D003BD"
2090 DATA "1AC09DCA07AD0ACF"
2100 DATA "D005BD17C0D003BD"
2110 DATA "1AC09DD707A9019D"
2120 DATA "ADDB9DCADB9DD7DB"
2130 DATA "CAD0CA6020FAC1A9"
2140 DATA "5191FA2008C2B1FC"
2150 DATA "8D14CF91FC6020FA"
2160 DATA "C1A92E91FA2008C2"
2170 DATA "AD14CF49FF31FC91"
2180 DATA "FC6078AD11D0297F"
2190 DATA "8D11D0A9008D12D0"
2200 DATA "AD1AD009018D1AD0"
2210 DATA "A9EA8D1403A9C38D"
2220 DATA "1503AD0EDC29FE8D"
2230 DATA "0EDC5860AD19D009"
2240 DATA "018D19D068A868AA"
2250 DATA "6840A9012C19D0F0"
2260 DATA "F3AD08CF8D21D0AD"
2270 DATA "0ACFD004A900F002"
2271 REM ::::::::::::::::::::::::::::::::
2272 REM :                              :
2273 REM : BLOCK 4  ( $C400 - $C4FF ) :
2274 REM :                              :
2275 REM ::::::::::::::::::::::::::::::::
2280 DATA "A90F8D1CD0A204AD"
2290 DATA "0ECF9D26D0AD04CF"
2300 DATA "9DF707CAD0F1A900"
2310 DATA "8D12D04CDCC32008"
2320 DATA "C2A000B1FC49FF91"
2330 DATA "FCC8C03FD0F54C8C"
2340 DATA "C1640A01EAEE09CF"
2350 DATA "60EE0ECF60EE25D0"
2360 DATA "60EE26D060AD0FCF"
2370 DATA "49018D0FCF4C53C3"
2380 DATA "AD0ACF49018D0ACF"
2390 DATA "4C53C32008C2A000"
2400 DATA "9891FCC8C03FD0F9"
2410 DATA "4C8CC1AD8D022901"
2420 DATA "F003205BC42016C2"
2430 DATA "A9298D00CFA9048D"
2440 DATA "01CFA2018E02CF8E"
2450 DATA "03CFCA8E07CFA980"
2460 DATA "8D14CF60AD8D0229"
2470 DATA "0148208CC368D003"
2480 DATA "4C35C24CA8C2AD8D"
2490 DATA "022901D0062073C2"
2500 DATA "4CB6C420F7C24C9E"
2510 DATA "C3AD8D0229014820"
2520 DATA "9EC368D0034C35C2"
2530 DATA "4CA8C2AE13CFCAD0"
2540 DATA "01608E13CF60AE13"
2550 DATA "CFE8E001D0F46020"
2560 DATA "75C1A000B1FD9920"
2570 DATA "CFC8C03FD0F66020"
2580 DATA "75C1A000B920CF91"
2590 DATA "FDC8C03FD0F64C8C"
2591 REM ::::::::::::::::::::::::::::::::
2592 REM :                              :
2593 REM : BLOCK 5  ( $C500 - $C5FF ) :
2594 REM :                              :
2595 REM ::::::::::::::::::::::::::::::::
2600 DATA "C1AE04CFE8D00160"
2610 DATA "8E04CFA203BD0ACF"
2620 DATA "186901C93AD009A9"
2630 DATA "309D0ACFCAD0EE60"
2640 DATA "9D0ACF2053C320DF"
2650 DATA "C44C8CC1AE04CFCA"
2660 DATA "E07FD001608E04CF"
2670 DATA "A203BD0ACF38E901"
2680 DATA "C92FD0DCA9399D0A"
2690 DATA "CFCAD0EE602008C2"
2700 DATA "A215A000B1FC0A91"
2710 DATA "FCC8B1FC0A91FC90"
2720 DATA "0888B1FC090191FC"
2730 DATA "C8C8C003D0ECA5FC"
2740 DATA "18690385FCA5FD69"
2750 DATA "0085FDCAD0D44C8C"
2760 DATA "C12008C2A215A002"
2770 DATA "B1FC4A91FC88B1FC"
2780 DATA "4A91FC9008C8B1FC"
2790 DATA "098091FC8888C0FF"
2800 DATA "D0ECA5FC18690385"
2810 DATA "FCA5FD690085FDCA"
2820 DATA "D0D44C8CC12008C2"
2830 DATA "A214A5FC18693C85"
2840 DATA "FAA5FD690085FBA5"
2850 DATA "FA38E90385FCA5FB"
2860 DATA "E90085FDA000B1FC"
2870 DATA "91FAC8C003D0F7A5"
2880 DATA "FA38E90385FAA5FB"
2890 DATA "E90085FBCAD0D8A0"
2900 DATA "009891FAC8C003D0"
2910 DATĀ "F94C8CC12008C2A2"
2911 REM ::::::::::::::::::::::::::::::::
2912 REM :                              :
2913 REM : BLOCK 6  ( $C600 - $C6FF ) :
2914 REM :                              :
2915 REM ::::::::::::::::::::::::::::::::
2920 DATA "14A5FC18690385FA"
2930 DATA "A5FD690085FBA000"
2940 DATA "B1FA91FCC8C003D0"
2950 DATA "F7A5FC18690385FC"
2960 DATA "A5FD690085FDCAD0"
2970 DATA "D8A0009891FCC8C0"
2980 DATA "03D0F94C8CC1A200"
2990 DATA "8A9D60CF9DA8CFE8"
3000 DATA "E048D0F52008C2A9"
3010 DATA "6085F8A9CF85F9A0"
3020 DATA "00B1FC91F8C8C03F"
3030 DATA "D0F7A9008502A203"
3040 DATA "8A48A98085FEA208"
3050 DATA "8A48A9ED85FAA9CF"
3060 DATA "85FBA0009848B1F8"
3070 DATA "A2080A489008A402"
3080 DATA "B1FA05FE91FAA5FA"
3090 DATA "38E90385FAA5FBE9"
3100 DATA "0085FB68CAD0E368"
3110 DATA "A8C8C003D0D646FE"
3120 DATA "A5F818690385F8A5"
3130 DATA "F9690085F968AACA"
3140 DATA "D0B6E60268AACAD0"
3150 DATA "A7A000B9A8CF91FC"
3160 DATA "C8C03FD0F64C8CC1"
3170 DATA "2008C2A5FC18693C"
3180 DATA "85FAA5FD690085FB"
3190 DATA "A20AA000B1FC48B1"
3200 DATA "FA91FC6891FAC8C0"
3210 DATA "03D0F1A5FC186903"
3220 DATA "85FCA5FD690085FD"
3230 DATA "A5FA38E90385FAA5"
```




```
3231 REM ::::::::::::::::::::::::::::::::
3232 REM :                              :
3233 REM : BLOCK 7  ( $C700 - $C7FF ) :
3234 REM :                              :
3235 REM ::::::::::::::::::::::::::::::::
3240 DATA "FBE90085FBCAD0D2"
3250 DATA "4C8CC1AE02CFCA86"
3260 DATA "FBA90385FC205BC1"
3270 DATA "A6FD2008C28AA8A2"
3280 DATA "03A50291FCC8CAD0"
3290 DATA "FA4C8CC1AE03CFCA"
3300 DATA "8A29F84A4A4A4820"
3310 DATA "08C268A8A215B1FC"
3320 DATA "0D14CF91FCA5FC18"
3330 DATA "690385FCA5FD6900"
3340 DATA "65FDCAD0E94C8CC1"
3350 DATA "AE03CFCA8A29F84A"
3360 DATA "4A4A482008C268A8"
3370 DATA "A215AD14CF49FF31"
3380 DATA "FC91FCA5FC186903"
3390 DATA "85FCA5FD690085FD"
```

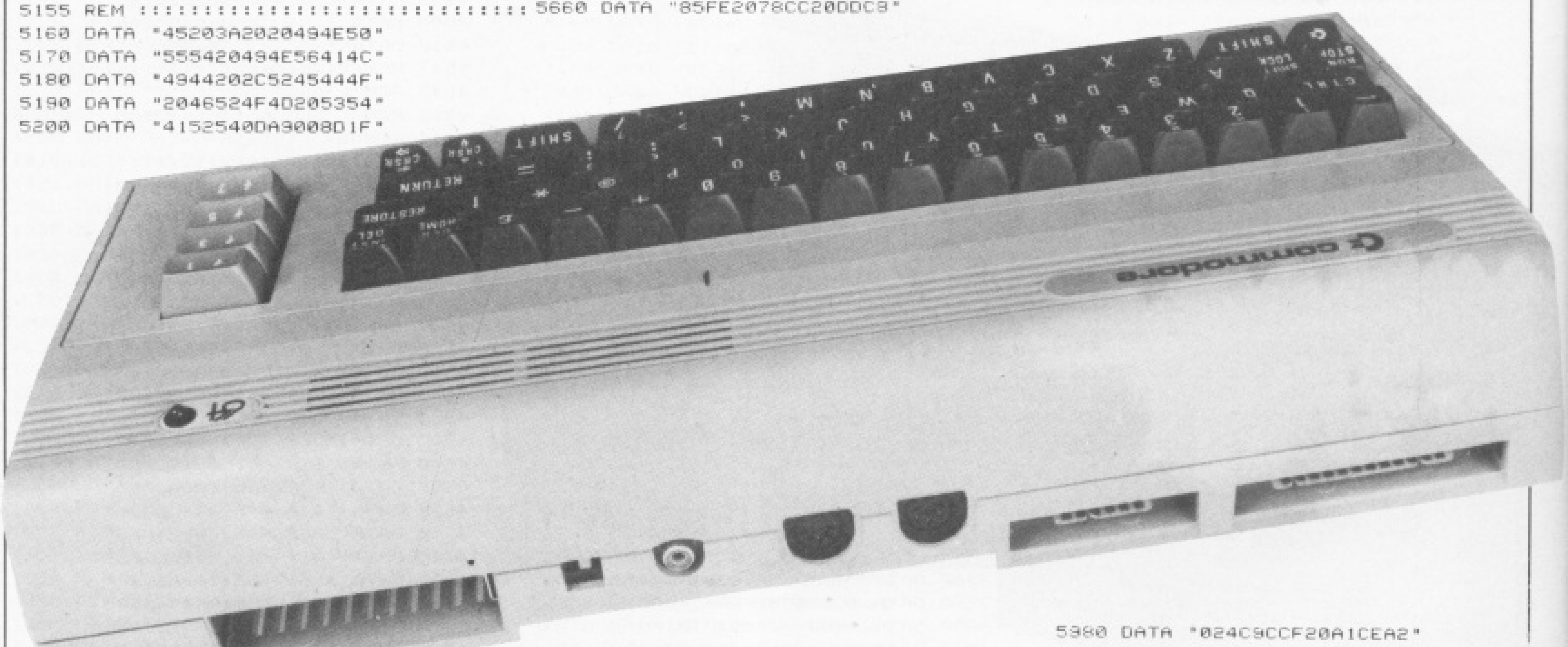
Program listing

```
3400 DATA "CAD0E74C8CC1A900"
3410 DATA "85024C0BC7A9FFD0"
3420 DATA "F7AD8D022901D094"
3430 DATA "4C8DC7AD8D022901"
3440 DATA "D0B64C86C72008C2"
3450 DATA "A000A9008502B1FC"
3460 DATA "A2084A48900AA502"
3470 DATA "0A090185024CC2C7"
3480 DATA "060268CAD0ECA502"
3490 DATA "91FCC8C03FD0DB60"
3500 DATA "2008C2A000B1FC48"
3510 DATA "4A4A4A4A4A4A8502"
3520 DATA "68480A0A0A0A0A0A"
3530 DATA "1865028502684829"
3540 DATA "304A4A1865028502"
3550 DATA "68290C0A0A186502"
3551 REM :::::::::::::::::::::::::::::::
3552 REM :                             :
3553 REM : BLOCK 8  ( $C800 - $C8FF ) :
3554 REM :                             :
3555 REM :::::::::::::::::::::::::::::::
3560 DATA "91FCC8C03FD0CE60"
3570 DATA "AD0ACFD00620A5C7"
3580 DATA "4C16C820D0C7A215"
3590 DATA "A000B1FC48A002B1"
3600 DATA "FCA00091FCA00268"
3610 DATA "91FCA5FC18690385"
3620 DATA "FCA5FD690085FDCA"
3630 DATA "D0DE4C8CC1A901A8"
3640 DATA "AE10CF20BAFFAD12"
3650 DATA "CFA2F0A0CF4CBDFF"
3660 DATA "A0009820CFFFC90D"
3670 DATA "F00899F0CFC8C010"
3680 DATA "D0F18C12CF6078AD"
3690 DATA "19D009018D19D0A9"
3700 DATA "008D1AD0A9318D14"
3710 DATA "03A9EA8D1503A947"
3720 DATA "8D1803A9FE8D1903"
3730 DATA "A90085C64CE8CC00"
3740 DATA "2904010180002000"
3750 DATA "0601003132380100"
3760 DATA "0100002C80000000"
3770 DATA "A218BD8FC89DFFCE"
3780 DATA "CAD0F760243A2462"
3790 DATA "248A24CAA208BDB3"
3800 DATA "C89DFFCFCAD0F7A9"
3810 DATA "0C9D17D0A90A8D1D"
3820 DATA "D0A90F8D10D08D15"
3830 DATA "D060203DC3AC13CF"
3840 DATA "A200CAD0FD88D0F8"
3850 DATA "60AD04CF8D60CFA2"
3860 DATA "03BD0ACF9D60CFCA"
3870 DATA "D0F720DAC8209FFF"
3871 REM :::::::::::::::::::::::::::::::
3872 REM :                             :
3873 REM : BLOCK 9  ( $C900 - $C9FF ) :
3874 REM :                             :
3875 REM :::::::::::::::::::::::::::::::
3880 DATA "A5C5C901F039C93F"
3890 DATA "F021C928D0062001"
3900 DATA "C54C1BC9C92BD003"
3910 DATA "202CC5A203BDAD07"
3920 DATA "49809DAD07CAD0F5"
3930 DATA "4CFAC8AD60CF8D04"
3940 DATA "CFA203BD60CF9D0A"
3950 DATA "CFCAD0F74C23C520"
3960 DATA "08C2A5FC85C1A5FD"
3970 DATA "85C2202BC920E4CB"
3980 DATA "4C23C5A000B1FC91"
3990 DATA "C1C8C03FD0F760A0"
4000 DATA "00B1FC48B1C191FC"
4010 DATA "6891C1C8C03FD0F1"
4020 DATA "60A000B1C111FC91"
4030 DATA "C1C8C03FD0F560A2"
4040 DATA "53A9C98E2E038D2F"
4050 DATA "034CE9C8A25FA9C9"
4060 DATA "D0F1A271A9C9D0EB"
4070 DATA "78A9A58D1803A9C9"
4080 DATA "8D19035860EA488A"
4090 DATA "483848EE08CF68A8"
4100 DATA "68AA684000A96885"
4110 DATA "FA85FCA90585FBA9"
4120 DATA "D985FDA208A001B1"
4130 DATA "FA48B1FC8891FC68"
4140 DATA "91FAC8C8C028D0EF"
4150 DATA "A5FA18692885FA85"
4160 DATA "FCA5FB690085FB69"
4170 DATA "D485FDCAD0D760AD"
4180 DATA "0EDC29FE8D0EDCA5"
4190 DATA "0129FB8501AEB4C9"
4191 REM :::::::::::::::::::::::::::::::
4192 REM :                             :
4193 REM : BLOCK 10 ( $CA00 - $CAFF ) :
4194 REM :                             :
4195 REM :::::::::::::::::::::::::::::::
4200 DATA "BD97CA85FBA90885"
4210 DATA "FC205BC1A5FE1869"
4220 DATA "D085FEA000B1FD99"
4230 DATA "E807C8C008D0F6A5"
4240 DATA "0109048501AD0EDC"
4250 DATA "09018D0EDC60CEF0"
4260 DATA "07D019A9088DF007"
4270 DATA "AEB4C9E8E0B0D002"
4280 DATA "A2008EB4C920EFC9"
4290 DATA "4C2ECA20B5C9A98F"
4300 DATA "85FA85FCA90585FB"
4310 DATA "A9D985FDA000A200"
4320 DATA "8A48BDE8070A9DE8"
4330 DATA "07900BA201A96C91"
4340 DATA "FA8A91FCD006A206"
4350 DATA "A920D0F3A5FA1869"
4360 DATA "2885FA85FCA5FB69"
4370 DATA "0085FB69D485FD68"
4380 DATA "AAE8E008D0CA6020"
4390 DATA "2010120513050E14"
4400 DATA "090E0720270D2E0F"
4410 DATA "2E022E200D010B05"
4420 DATA "122D3634272C0405"
4430 DATA "1309070E05042001"
4440 DATA "0E04201712091414"
4450 DATA "050E20060F122027"
4460 DATA "190F151220030F0D"
4470 DATA "0D0F040F1205272C"
4480 DATA "010E200112071513"
4490 DATA "201310050309010C"
4500 DATA "091314201015020C"
4510 DATA "09030114090F0E2C"
4511 REM :::::::::::::::::::::::::::::::
4512 REM :                             :
4513 REM : BLOCK 11 ( $CB00 - $CBFF ) :
4514 REM :                             :
4515 REM :::::::::::::::::::::::::::::::
4520 DATA "0219200A0F080E20"
4530 DATA "0D032008010C0520"
4540 DATA "1820280329200E0F"
4550 DATA "16050D0205122031"
4560 DATA "393834201D2E2010"
4570 DATA "1205131320271205"
4580 DATA "1415120E2720140F"
4590 DATA "2013140112142E20"
4600 DATA "2E202E202E202E20"
4610 DATA "DAC8209FFFA5C5A2"
4620 DATA "00DD70CBD00FBD90"
4630 DATA "CB8D2E03BDB0CB8D"
4640 DATA "2F036C2E03E8E020"
4650 DATA "D0E760EAEAEAEA60"
4660 DATA "3302072C003C0924"
4670 DATA "383B393530282B20"
4680 DATA "2111030604052D32"
4690 DATA "15122E31362A0D3E"
4700 DATA "6B6CF094A6B94550"
4710 DATA "3D413935EF012C1E"
4720 DATA "C832814DFCB5CBD6"
4730 DATA "919B7F8C92AFB7EA"
4740 DATA "C4C2C2C4C4C4C4C4"
4750 DATA "C4C4C4C4C4C5C5C4"
4760 DATA "C6CCC5C5C5C5C4C4"
4770 DATA "C7C7C9C9C9CECECB"
4780 DATA "A99320D2FF2054C0"
4790 DATA "208CC12053C320B2"
4800 DATA "C34CBCC82008C26C"
4810 DATA "2E032066C8A2008A"
4820 DATA "9D00D0E8E010D0F8"
4830 DATA "8D15D08D10D08D1D"
4831 REM :::::::::::::::::::::::::::::::
4832 REM :                             :
4833 REM : BLOCK 12 ( $CC00 - $CCFF ) :
4834 REM :                             :
4835 REM :::::::::::::::::::::::::::::::
4840 DATA "D08D17D08D0FCC20"
4850 DATA "D2FFE8E01CD0F5A9"
4860 DATA "0E8D86028D20D049"
4870 DATA "088D21D04C2BCC93"
4880 DATA "098E055155495420"
4890 DATA "2E0D0DA90085C66C"
4900 DATA "0203AD8D022901D0"
4910 DATA "034C08C84C36C6A7"
4920 DATA "B1A7A0BDA0948190"
4930 DATA "85ACA7B8A7A0BDA0"
4940 DATA "3489938BACA79295"
4950 DATA "8EAF93948F90A7A0"
4960 DATA "BDA08292858188A2"
4970 DATA "28BD3ECC9DFF03A9"
```



Program listing

```
4980 DATA "019DFFD7CAD0F260"
4990 DATA "A9F7A6FDA4FE4CD8"
5000 DATA "FF204C4F41442053"
5010 DATA "5052495445204441"
5020 DATA "54412046524F4D20"
5030 DATA "4445564943452E0D"
5040 DATA "0D20534156452053"
5050 DATA "5052495445204441"
5060 DATA "544120544F204445"
5070 DATA "564943452E20200D"
5080 DATA "0D46524F4D544F20"
5090 DATA "2020424C4F434B20"
5100 DATA "4E554D424552203F"
5110 DATA "2020454E54455220"
5120 DATA "46494C454E414D45
5130 DATA "8D15D0AD0EDC0901"
5140 DATA "8D0EDC5860000000"
5150 DATA "00000000204E414D"
5151 REM :::::::::::::::::::::::::::::::
5152 REM :                             :
5153 REM : BLOCK 13 ( $CD00 - $CDFF ) :
5154 REM :                             :
5155 REM :::::::::::::::::::::::::::::::
5160 DATA "45203A2020494E50"
5170 DATA "555420494E56414C"
5180 DATA "4944202C5245444F"
5190 DATA "2046524F4D205354"
5200 DATA "4152540DA9008D1F"
```



```
5210 DATA "CF8D18CF8D19CFA0"
5220 DATA "039914CF88D0FA20"
5230 DATA "CFFFC920F0F9C90D"
5240 DATA "F00A9915CFC8C003"
5250 DATA "D0EDA90D20D2FF20"
5260 DATA "D2FFB914CFC93090"
5270 DATA "32C93AB02E38E930"
5280 DATA "85FBB930C485FC20"
5290 DATA "5BC1AD18CF1865FD"
5300 DATA "8D18CFAD19CF65FE"
5310 DATA "8D19CF88D0D4AD19"
5320 DATA "CFD008AD18CFC980"
5330 DATA "900160A9018D1FCF"
5340 DATA "602066C8A9068D21"
5350 DATA "D0A90E8D20D08D86"
5360 DATA "02A99320D2FFA90D"
5370 DATA "20D2FF20D2FFA200"
5380 DATA "AD11CFD0058D81CC"
5390 DATA "D003BDA1CC20D2FF"
5400 DATA "E8E020D0EBAD11CF"
5410 DATA "D0034C18CEA200BD"
5420 DATA "C1CC20D2FFE8E004"
5430 DATA "D0F5BDC5CC20D2FF"
5440 DATA "E8E014D0F52024CD"
5450 DATA "AD1FCFF00620A1CE"
5460 DATA "4CCDCDAD18CF85F7"
5470 DATA "A200BDC5CC20D2FF"
5471 REM :::::::::::::::::::::::::::::::
5472 REM :                             :
5473 REM : BLOCK 14 ( $CE00 - $CEFF ) :
5474 REM :                             :
5475 REM :::::::::::::::::::::::::::::::
5480 DATA "E8E014D0F52024CD"
5490 DATA "AD1FCFF00620A1CE"
5500 DATA "4CF8CDAD18CF85F8"
5510 DATA "A200BDD9CC20D2FF"
5520 DATA "E8E00FD0F5A90D20"
5530 DATA "D2FF20D2FFA200BD"
5540 DATA "FCCC20D2FFE8E008"
5550 DATA "D0F52050C8A90D20"
5560 DATA "D2FF2067CCA5C5C9"
5570 DATA "3FF04DC938D004A9"
5580 DATA "01D006C91BD0EEA9"
5590 DATA "088D10CF203DC8AD"
5600 DATA "11CFD00620D5FF4C"
5610 DATA "95CEA5F7A4F885FC"
5620 DATA "A94085FB205BC1A5"
5630 DATA "FD85F7A5FE85F898"
5640 DATA "85FC205BC1A5FD18"
5650 DATA "694085FDA5FE6900"
5660 DATA "85FE2078CC20DDC8"
5670 DATA "2098C920DFC44CD0"
5680 DATA "CBA200BD04CD20D2"
5690 DATA "FFE8E020D0F560A9"
5700 DATA "008D11CF4C91CDA9"
5710 DATA "01D0F620A8C8A98E"
5720 DATA "8D20D020D2FFA906"
5730 DATA "8D21D0A90820D2FF"
5740 DATA "A99320D2FFA9018D"
5750 DATA "F007A9FF8DB4C9A9"
5760 DATA "188D13CF202ECA20"
5770 DATA "DDC8A5C5C901D0F4"
5780 DATA "68682098CE20DDC8"
5790 DATA "2047CB4CF5CE0000"
5791 REM :::::::::::::::::::::::::::::::
5792 REM :                             :
5793 REM : BLOCK 15 ( $CF00 - $CFFF ) :
5794 REM :                             :
5800 DATA "A90E8D20D08D21D0"
5810 DATA "A9008D3602A200BD"
5820 DATA "1DCF20D2FFE8E048"
5830 DATA "D0F54C66CF938E08"
5840 DATA "1120454E54455220"
5850 DATA "444556494345204E"
5860 DATA "4F2E202620505245"
5870 DATA "5353202752455455"
5880 DATA "524E272E0D0D2031"
5890 DATA "3D5441504520 3A20"
5900 DATA "383D4449534B2E0D"
5910 DATA "0D20444556494345"
5920 DATA "203A203F2000A900"
5930 DATA "8D65CF20CFFFC920"
5940 DATA "F0F9C90DF0058D65"
5950 DATA "CFA90D20D2FF20D2"
5960 DATA "FFAD65CFC930900C"
5970 DATA "C93AB00838E93085"
5980 DATA "024C9CCF20A1CEA2"
5990 DATA "3A4C0FCFA502C901"
6000 DATA "D0034CA9CFC908D0"
6010 DATA "EBAAA90D20D2FF20"
6020 DATA "D2FFA92020D2FFA9"
6030 DATA "0085FAA9C085FBA9"
6040 DATA "01A00320BAFFA904"
6050 DATA "A2DDA0CF20BDFFA9"
6060 DATA "FAA200A0CF20D8FF"
6070 DATA "A90085C8604D4D36"
6080 DATA "3400000000000000"
6090 DATA "0000000000000000"
6100 DATA "0000000000000000"
6110 DATA "0000000000000000"
9999 DATA "END"
10000 POKE53281,0:POKE53280,0
```

```
10010 T$(1)="PROGRAM NAME : 'M.O.B MAKER-64'"
10020 T$(2)="AUTHOR       : JOHN MC HALE    "
10030 T$(3)="WRITTEN FOR  : 'YOUR COMMODORE'"
10040 T$(4)="RELEASE DATE : NOVEMBER 1984   "
10060 PRINT"■":REM [CLR/HOME]
10070 FORC=1TO4
10080 L=LEN(T$(C))
10090 FORD=1TOL
10100 PRINTMID$(T$(C),D,1);
10110 FORTM=0TO59:NEXTTM
10120 NEXTD
10140 PRINT:PRINT
10150 NEXTC
10160 PRINT:PRINT:PRINT"■■ PLEASE WAIT - HEX CONVERSION WILL TAKE ■"
10170 PRINT"■ SEVERAL MOMENTS . . . . . . . . . . . . .■"
10180 RETURN
```

INPUT

If you've got any useful advice to give Your fellow Commodore readers or any problems with which you need our experts' help, put pen to paper.



INPUT

With reference to Concorde II (volume 1, issue 1), to make the program respond correctly to the LEFT and RIGHT, PORT and STARBOARD movement of the joystick in Port 2, the following lines should surely be:

```
80  IF(AAAND8)ANDTT=2 ......
81  IF(AAAND4)ANDTT=2 ......
82  IF(AAAND8)ANDTT=3 ......
83  IF(AAAND4)ANDTT=1 ......
```

Despite this, it really is a super program.
John Wilkes
Dursley

INPUT

Two years ago, the Computer Press were telling us all how good the VIC 20 was with its excellent graphics and sound capabilities. Now, with a few exceptions, there is absolute silence from the press – so, what has changed? The answer is nothing: the VIC is still the same excellent micro, offering even better value now with a reduced price.

What will really "kill off" the VIC is if new, and established, owners can no longer find programs, hints and articles for it. Apart from continuing to print articles like 'VIC Games Programming' and games, perhaps Your Commodore could apply subtle pressure to manufacturers. For example, if an item or program is sent in for review, Your Commodore could ask if it is suitable or available for the VIC.

Perhaps a VIC 'special' could be planned, asking software houses to send details of VIC programs for inclusion in a mammoth listing.

By reminding manufacturers and dealers that a large number of Your Commodore readers have VICs, all eager to part with their money for quality programs, you would be helping to keep the VIC a worthwhile proposition.

So, keep up the good work with Your Commodore but make it even better by keeping the VIC alive and thriving.
Bob Black
London

I want to interface an Epson FX80 Printer to my 64. I should like the Epson to behave exactly like a Commodore printer and the interface to be 'user transparent'. I had considered the 'Interpod' but note that it hasn't got a centronics interface. Is there any disadvantage in the optional RS-232C interface available on the Epson?
Hugh Hennessy
Co. Antrim

OUTPUT

No matter which interface you use for your FX80 Printer, it will never be totally compatible with a Commodore Printer. This is because all Commodore equipment (for reasons best known to themselves) use their own code for storing characters, and not ASCII which most other peripherals (including the FX80) use. For example the 64 stores the letter 'A' as 65 but the ASCII for 'A' is 97. This means that if you require upper and lower case letters to work correctly all characters sent to the printer will have to be converted to ASCII.

This can be very time consuming but some interfaces do this conversion for you and are well worth the extra cost. However no non-Commodore printer will be able to print the comprehensive range of graphics characters included in the 64's character set.

As for the choice between centronics or RS232 interfaces, it really is swings and roundabouts. Centronic interfaces require more wires, while RS232 interfaces require setting up of Baud Rates, paraty, stop bits etc. However the 64 does already have the firmware for handling RS232 output on the user port.

INPUT

The Commodore recorder specifies that tapes of less than 30 minutes must be used (I assume this is 15 minutes each side). The majority of tapes available are either 10 minutes or 15 minutes long.

Which do I choose for general use? It would also help if programs printed in your magazine gave some idea of the length of tape required.
R. Marks
Gloucestershire

OUTPUT

For most programs that are printed in magazines a C12 will be more than sufficient. You will only need a longer tape if you are writing a very long program or one using a large amount of text.

INPUT

I am learning machine code and am having problems in getting the random function to work. Please could you help me as I can't find the answer in any books or magazines.
W. Laing
Lanarkshire

OUTPUT

The random number used by basic is stored in locations 8B — 8F inclusive. To generate the next random number simply call the routine at $E097.

INPUT

When using machine code to perform arithmetical calculations, having obtained a numerical answer to a series of additions, how do you print out the answer to screen or printer? To take the simplest case, supposing the answer is # $EFF(=255), which is held in the accumulator, and the next instruction is the 'output' instruction, JSR$FFD2, the

machine will print the ASCII version of 255 which is the symbol ' '. But how can I get the machine to print the actual number – i.e. 255. I understand how to do it if the arithmetic is in Binary Coded Decimal or if I store the result in an address, revert to BASIC and use a 'PRINT PEEK (HIGH BYTE) 256 PEEK (LOW BYTE) instruction; but how do I do it directly?
M.W. Peters
Dorset

OUTPUT

Most numbers that you will want to print out will be 16 bit. So here is a routine that will print out a number passed to it in the accumulator and the X register (A=LO BYTE,X= HIGH BYTE). For 8 bit numbers just set X=0 before calling it.

```
e.g. To print a number stored in point
     LDA Point
     LDX Point + 1
     JSR STR
or   To print 255
     LDA # $FF
     LDX # $00
     JSR STR
```

INPUT

With the demise of the VIC 20 there may be a number of your readers who are considering the 64 as a replacement. Let me sound a note of caution. My original configuration was the VIC 20, 1515 Printer, Datasette and 1540 Disc Drive. On enquiry from the supplier I was assured they were all compatible with the 64. The supplier gave me a weird 'Open' command to use with the Disc Drive which was confirmed by CBM Corby. Needless to say it did not work. Further enquiry to CBM gave me a couple of 'Pokes'. This appeared to work until I attempted loading a database program. Yet another enquiry gave me the information that the 'Pokes' would not work if there were any 'Loads'/'Saves' in the program and the only way to ensure success was to change a chip in the 1540. I loaded the 64 magazine tape from the Datasette and got a 60% 'Load Error' response. A friend loaned me his C2N and everything was perfect. This means I've spent £22.42 for a chip and £39.95 for a C2N. So, when a supplier tells you that the peripherals are 64 compatible – they're not.
C.K.R. Harris
Fareham



INPUT

I am considering the purchase of a colour monitor and have not been very impressed with the Commodore 1701 when compared with, say, a BBC'B' with a Microvitec CUB. Is this a feature of the Composite Video standard when compared with RGB, or have I just seen bad examples?

Is there any harm in opting for one of the combined TV/Monitors that are coming onto the market? I have only seen one in the flesh and it was being run as a TV; it appeared to have a single BNC input for the video signal with a single PHONO connector for the sound (this was a FISHER 1401). I have also seen advertised the SAISHO CM20R, FERGUSON TXMC10 and the FIDELITY CTM1400. These seem to need a more comprehensive set of inputs – which is what I would expect.

If the results are comparable with the 1701, then I would gladly pay the extra £50 or so on the discount price to acquire a second TV.

Please could you advise me.
R.J.K. Murphy
West Lothian

OUTPUT

As you suspect the drop in quality between the BBC you saw running and the 64 with the 1701, is due to the use of the composite video link. This is because with composite video, the three primary colour signals generated in the 64 have to be combined into one, and subsequently decoded by the monitor. It's the process of encoding and decoding the video information that results in the quality drop. The picture obtained from any of the monitors you mention will certainly give as good an image as the 1701. However, for only a slight drop in contrast any good small screen television, if correctly tuned, will give a comparable picture. This does rely on the tuning of the RF converter in the Commodore machines remaining stable which was not the case in early machines where the tuning tended to wander.

If you can afford the difference the combined TV/Monitors are your best bet. A bonus point being that they provide a compatible RF input for any micro.

INPUT

I have compiled several programs for musical tunes but only using one voice. Could you please tell me how two or three voices are programmed.

Also, although I can move a balloon sprite back and forth across the screen, followed by scrolling an aeroplane up and down, I cannot write a program enabling both to move on the screen at the same time and would appreciate your guidance.
J.S. Thomas
Torquay

OUTPUT

To program the other two voices on your 64 all you need to do is repeat the code for programming the first voice and add 7 to all the poke locations except that for volume. For example, to set the frequency of voice 1 the line might be:
POKE 54272, 100: POKE 54273, 10
To set voice 2 use
POKE 54279, 100: POKE 54280, 10
Be careful to trigger all three voices as closely together as possible otherwise the chords will become staggered.

To move more than one sprite at a time, include the pokes for moving both sprites within the same loop. For example, to move sprite 1 across the screen chased by sprite 2 use
For I 10 to 200: Poke 53250. I+40: Poke 53252, I:next

OUTPUT

COMP

You do not have to accept this mission but if you decide to there are some fantastic games to be won.

"Destroy him my robots!"

These chilling words may be the last you'll ever hear. They are spoken by the evil Elvin Atombender as you search his underground stronghold for the clues which will enable you to foil his plan to destroy the world. Lined up against you are his deadly robot guards. Your only weapon is your pocket computer with which you can store and analyse the information you collect.

If you decide to accept this mission you will need a copy of Impossible Mission from CBS Software. **Your Commodore**, always ready to help in the fight to save the world, has come to the rescue. We are offering ten lucky (?) people the chance to meet the evil Elvin. Perhaps you'll be the one to defy his threats, to overcome his robots, to find his hidden password and to rid the world of his wickedness. Or more likely, you'll just plunge to your death with a blood-curdling scream. Either way you'll experience a game with brilliant graphics, stunning sound effects, realistic speech and a fiendishly difficult plot.

The rewards

Of course, that's not all. The first prize winner will receive not only a copy of Impossible Mission but the complete CBS Software launch range. That's eight great titles in all.

There are two second prizes of Impossible Mission plus three other titles, two third prizes of Impossible Mission plus two other titles and four fourth prizes of Impossible Mission plus one other title.

That's a lot of software but we haven't finished yet. There are **fifty** (yes, fifty) runners-up prizes to be won. They will all receive one of the top titles listed opposite.

How to enter

To complete the mission we have set you is not at all impossible. However, you do have to find a password which will be a combination of the five letters by the robot pictures. To find the password, take a look at the five pictures and then match each robot to the film in which you think it appeared. For example, if you think that Robot C was in Star Wars, then C is the first letter of your password and so on.

If you're not sure, have a go anyway. With so many prizes you may still win even if you haven't got all the answers right.

Fill in your password, name and address onto the entry coupon and send it to **Impossible Mission Competition, Your Commodore, 1 Golden Square, London W1R 3AB.** The closing date for the competition is last post on Friday March 29, 1985.

Please write your password onto the back of the envelope in which you send your entry otherwise we will not be able to accept it.

You may enter as many times as you wish, but each entry must be on an official coupon – not a copy – and sealed in a separate envelope. Please write clearly on the coupon as it will be used as a label if you win a prize.

IMPOSSIBLE MISSION™





1. STAR WARS
2. DR. WHO AND THE DALEKS
3. THE BLACK HOLE
4. METROPOLIS
5. THE SHAPE OF THINGS TO COME





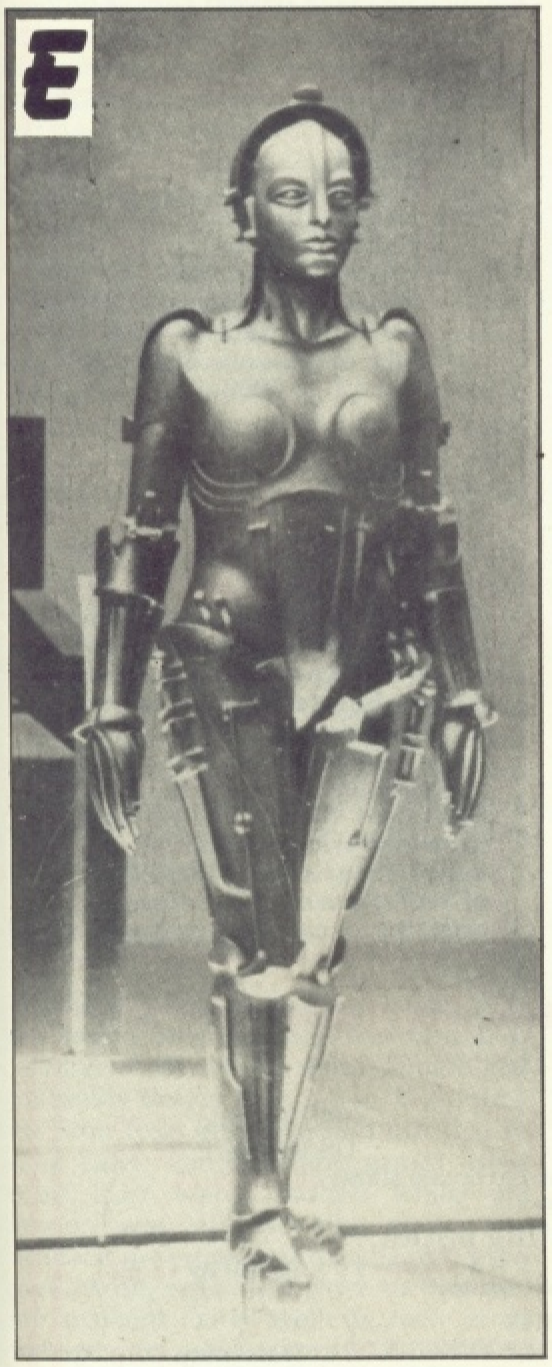
PETITION

The Rules

Entries will not be accepted from employees of Argus Specialist Publications Ltd, their printers and distributors, and CBS Software. This restriction also applies to employees' families and agents of the companies.

No correspondence will be entered into with regard to the competition results and it is a condition of entry that the editor's decision is final.

The How to Enter section forms part of the rules.



Great games up for grabs

WINNER TAKES ALL
The complete CBS Software range.

Impossible Mission
Destroy Elvin before his robots destroy you.

Breakdance
Try the latest dancing craze without breaking your neck.

Pit Stop
The right formula for fast action in the driving seat of racing car plus all the excitement of the pits.

Gateway to the Temple of Apshai
A classic adventure with priceless treasures (a good thing) and hungry monsters (which aren't)

Silicon Warrior
You are a Silicon Warrior in a 3D Power Grid. The action becomes faster as you try to program the chips in your grid.

Jumpman
Thirty different arcade games all with great graphics, colour sound. Jump for joy.

Dragonriders of Pern
An action/strategy game which blends battle music and breathtaking graphics.

Lunar Outpost
The world is threatened yet again, this time from outer space. Your job is to defend it from the last lunar outpost.

Impossible Mission Competition

I wish to be considered for this impossible mission, though I understand that it's really rather easy and my life certainly will not be endangered. I also realise that my copy of Your Commodore will not self-destruct in 30 seconds.

Name .... MR. D. M. LIVERSIDGE ....

Address .... 31 WARREN ROAD ....

.... WICKERSLEY, ROTHERHAM ....

SOUTH YORKSHIRE .... Postcode........

My password letters are .... ACDEB ....

Please tick whether you would prefer disc ☐ or cassette ☑
Remember – put your password letters on the back of the envelope or your entry is invalid.

David Crisp tests out a faster alternative to the standard tape — Entrepo's Waferdrive.

WAFERDRIVE



IF YOU HAVE HAD YOUR HEAD under a pillow for the last few months you may have missed all advertising and reviews on this type of device. Basically it is a tape storage device, rather like a standard cassette. However, this is much faster than a standard tape. The small cartridge, which is about the size of a box of matches, fits into a drive about one sixth of the size of a standard 1541. The whole thing then fits into the cassette port of your 64 (note not an SX-64 add-on). It is claimed that, due to the speed the flimsy looking tape runs at, 120K of data could be accessed in about 43 seconds. Sounds impressive but what was it like to use?

Once it is plugged in you initially have to load its operating system. Why does everybody use C000 to CFFF? This is where the operating system reposes and so does my software printer interface and many other useful bits and pieces. Turn it all off and start again. Right!

Nothing there

A separate tape was supplied with the drive. This claimed to contain games but, when I took a directory of the tape, I could only find something called script, which would not do anything except hang the whole thing up. This may have been due to somebody unconnected with Entrepo wiping whatever was on there to start.

There is a built in program which displays a menu on the screen enabling you to format, wipe, clean, copy wafer to wafer, wafer to tape, disc to wafer and so on as well as getting a directory of the tape. It is very slow. About 40 seconds seems to be the average time. It is a fair comment that as this is a tape it is not bad but I did find myself comparing things to a disc drive and even the dreaded 1541 is not as laboured as that.

Slow load

Regarding program loading times, it is very much between tape and disc. It is definitely faster than a standard tape and noticeably slower than a disc. But I did a few time tests and it turned out that in some cases standard tape games using turbo load were finished while the Entrepo was still whirring away. The manual is pretty good and explains how to open, close, read, and write to files etc. and explains fairly well its own error messages. In use it is much the same as cassette as all filing is sequential and is something you will find easy to use quite fast. The tapes themselves are resistant to most legal forms of abuse except for ovens, baths, and steamrollers as the tape part is covered by a sliding lid which protects them from all but the most persistent pokers, and this may be where they score.

Mix and match

It is possible to use two of the units together on one waferdrive and cassette. Because of power supply limitations that is all. Of course you can still use your disc drives as well.

The tapes are available in 5 lengths. These different lengths have different capacities and so with a shorter tape there is less searching for the drive to do resulting in faster loading etc. The tapes are as follows.

TAPE LEN ft	CAPACITY (max)	AV. Accs Time (in seconds)
10	15K	8
20	35K	15
35	65K	25
50	96K	34
62	120K	43

There are times as supplied by Entrepo. As more files are used so the capacity becomes less.

More commands

There are two types of file available — program and sequential. A maximum of 255 files of mixed type can go on each tape.

The following commands are relevant to Entrepo use: Open, Close, Get, Input#, Print#, (all relate to reading and writing to files), Load, Save, and verify. At first glance it may appear that a scratch command is missing but if you think about it for a while you will realise that due to the nature of the file a scratch command would be difficult to implement.

To initially use a tape it must first be formatted as for a disc. This is done by using the built in utility program as mentioned before.

Copy all

The copy routines provided make transferring files fairly easy. However, I found it to be unreliable. Also you had to specify whether a file to be copied was a program or sequential file. That is easy if you are copying from a disc but not so easy if you are copying an unknown piece of software from tape.

What it's all about

A Wafer drive sounds like something to eat but appears to be no more than a tape loop smaller than a standard cassette.

Personally there are not enough advantages for me to want one. I feel they may find their niche in computer circles but, and I may be wrong, I do not think they will catch on.

Your COMMODORE

YOUR BEST INDEPENDENT COMMODORE MAGAZINE

SO YOU OWN A COMMODORE?

SO YOU'VE WRITTEN SOME PROGRAMS?

SO WHY HAVEN'T YOU SUBMITTED THEM TO US?

Your Commodore is always on the lookout for new material for publication and we know that there are thousands of intelligent, literate, innovative and creative Commodore owners out there, so why don't we get together?

If you have written an exhilirating game or an invaluable utility on your Commodore micro, share your talents with us and our readers by submitting your efforts and the form to the address below. All articles should be documented and type-written and should be accompanied by a printout of the program as well as a copy of the program on cassette or disc. All material should be original; if it is not chosen for publication, it will be returned to you.

You may not have written any software yourself, but you have very firm opinions about the world of Commodore and all their attendant industries and products. Then put your opinions on paper and post them to us, again at the address below — you never know, you might even get paid for airing your views! All submissions should be sent to:

The Editor
Your Commodore
Argus Specialist Publications Limited
No 1 Golden Square
London WIR 3AB

★PLEASE COMPLETE IN BLOCK CAPITALS

Your Name

Program Name

Computer/memory size it runs on

Amount of memory program occupies

Other computers/memory size which your program runs on without conversion or use

Does your game need or use joysticks? Yes No

Have you sent your game to another magazine Yes No

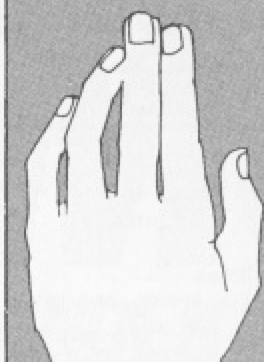
Is it original/or a variation on a theme?

Your Address

Telephone Number

Times to contact you

BUSINESS FILE



David Crisp assesses four business applications from Gemini software for the Commodore 64.

GEMINI HAVE BEEN IN THE SOFTWARE market for quite a while now. Most of their software seemed to start life on the BBC machine but is slowly being converted for use on others.

The first package I looked at in their serious software range was **Home Accounts.** This is intended to help you with your home budget. Personally I would have little use for a program of this type and feel that it is just one of those programs which you buy to show the spouse that computers can be used for serious applications! My wife wouldn't be fooled for a minute.

A program of this type requires discipline. Like a business program, if you do not keep it up to date you can find yourself in an awful mess.

After loading the program, either from tape or disc, you are asked if you have a file to load. Gemini thoughtfully provide some demonstration data in order to help you find your way around. You then have to say whether the data is on tape or disc. This is my first gripe. I feel that with this type of software, once is enough — whenever you save or load data you are asked the same question again. It would be nice if this information was saved after your first input. This does become annoying after a lot of use.

Looking through the manual I came across something I found hard to believe. Gemini warn you that Commodore's dreaded garbage collect routine may 'temporarily suspend program execution for up to two minutes' (their words not mine). What, thought I! Have they not even bothered to do something about it? Apparently not. According to Gemini this is unavoidable. It's not, and for a small fee I would show them how to avoid it.

Options

The main menu presents you with a comprehensive list of options. You are able to input the data which you feel you would spend on household items such as mortgage, insurance, rates, and so on. Another option then allows you to put in the actual figures as time goes by. You are able to put in bank standing orders, loans, and so on which build up into a set of figures which can be displayed on screen as a bar chart or printed out. The figures can be either one set of data only e.g. mortgage, or your whole budget. I must say that a bar chart is easier to digest and compare than simply a list of comparative figures.



The budget items that are provided seem to cover most things but you can change your headings if the need arises. All the things which you need to perform household accounts are there. It is as good as any of the other home account packages that I have seen although I feel there are a couple of areas in which it could be improved.

Stock control

The next Gemini program that was loaded into the 64 was their **Stock Control** program. This is the same in principle to earlier versions that I have seen, but seems to have been more effectively programmed. The whole thing is more professional and small bits and pieces such as a non-standard cursor flash give it a 'nice to use feel'. I enjoyed using this although I feel there are a couple of small things missing.

The one thing it does not offer is capacity. Apparently, there is a limit of 235 records per file. Of course it is possible to have more than one file but I feel that would make management fairly difficult. Some business users may find this adequate but even for my humble business, 235 cards are not sufficient.

Adding records is easy and most things are so simple it is almost possible to use without reading the manual although that is bad practice. To exit options it is usually necessary to press the 'home' key. That is alright in itself but it seems to differ from Gemini to Gemini program. A little consistency would help here.

If you just want to browse through the stock records you can and if you are looking for a specific record, there is either a search by stock number or search by datafield. This performed adequately but tended to be a little on the slow side.

Records can be sorted on any of the fields as long as there are at least 3 records.

Reports

One of the reports on the stock file that can be printed is a financial summary. This will break down a specified block of stock and show total costs of stock, retail value of stock, and the overall profit margin as well as the cost of bringing all understocked items up to minimum levels. Another, the stock summary, will again show the details from a specified block and either display or print parts of the selected records. It is also possible to get hardcopy of complete stock cards.

Printing

The whole program is orientated towards printing out on a Commodore printer but, if you have another type driven by a software interface, Gemini do at least suggest where it is possible to locate it (in memory fool). When printing out Gemini also show you how to customise the program in order to get your printer to print pound signs instead of a hash and how to print in upper as opposed to lower case. Thoughtful little pieces like this make the program much friendlier to use.

Omissions

The one thing I would like to have seen was an easier way to enter sold stock. As it stands you need to use the standard amend routine and this is a little long winded. Another thing which would have made it 'more useful in the field' would be a report of daily sales.

On the whole it is a considerable improvement on some of the Gemini programs I have seen but it is not one I would choose to use myself.

The Gemini **Database** is again rehashed from another machine but is, like the stock control, a considerable improvement on their earlier programs.

It is a stand alone database, by which I mean it is not programmable — but then for its price that is not unreasonable.

Once the program has loaded, the first thing you need to do, if you are not loading a previous set of records, is to format your record card. This is chosen from the main menu and is simple to do.

The documentation is clear and precise and there is good use of keys. For instance to change screen colours they have chosen to use the function keys. When your file loads next time the 64 will default to the colours you have chosen.

Each field you define can be up to 78 characters in length and you can have up to 20 fields per card. The number and size of the fields determines the total capacity of your file. Again, as with the other Gemini programs, sequential filing has been used and this tends to limit the total capacity of the file.

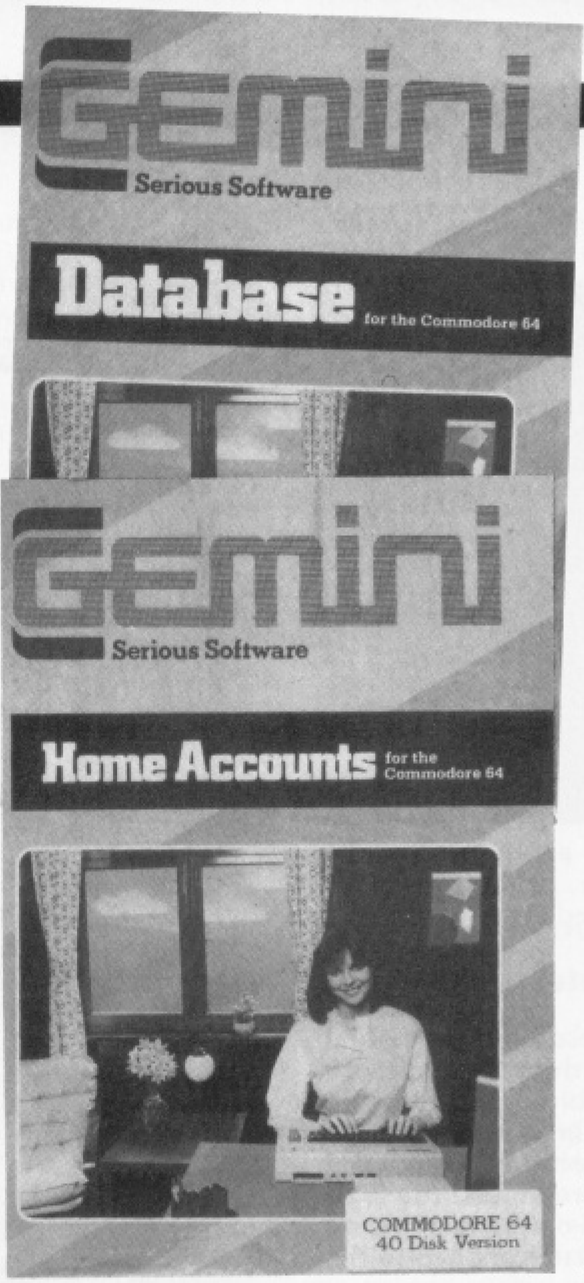
Getting filled in

Once you have finished formatting the card you can start to fill in. This is easy and, so as not to make your 64 throw a wobbly, Gemini have disabled certain keys which can produce the dreaded 'Extra ignored' error message which most people have reason to curse at times. Once you have some data in you can start to use it.

Every Gemini program seems to have its saviour and this one is no exception. The calculate feature in this one is the one that I find very useful. You can perform many calculations on any numeric field which can allow you to find such gems of knowledge as 'the total age of all your friends'. Seriously though it is a powerful function and one which I think many people may find useful.

Poking about

The search feature on this database is nice and works very well. Using things such as ><= and so on it is possible to find records which match a pretty wide set of data. The display at the top of the screen will show you how many records it has found



matching the conditions you have chosen.

Sort it and save it

A powerful sort option is provided which will allow you to sort on any field whether numeric or string and does offer case discrimination. It's a shame that the case discrimination was not present in the stock control.

As well as the expected load and save option there is also an append. This will allow you (memory permitting) to add another file onto the one currently in RAM. It does not matter if the card format is different but the format currently in memory takes precedence and the appended file fits itself into the presiding format. I found this very useful.

Printing out

Once again the report facilities are good. They can either go to the screen or printer and you can be specific as to which fields are output. You can also customise the program in order to get the best printout possible.

As with the stock program, the manual is full of tips on how to get your printer functioning properly and, on this one, an area of memory has been set aside specifically for a software interface. This is 500 bytes and is a 37579 to 37079.

This is a useful database and for the price is good value. It is a new version of an older program which was a total bodge up. I am glad to see it has been re-written and re-written effectively. Gemini seem to be getting better as time goes on.

The Gemini mail list is in effect very similar to the database program, except you use preformatted records. Certain functions such as calc are not required on this type of program and are indeed absent.

Search by key

Basically it is a name and address book which is used for printing out labels. Names and addresses are entered on a set form which as an extra, has a field called searchkey. This searchkey field allows you to input up to 10 character in order to designate points of note regarding this particular card.

For instance, if the file was of business contacts, key letter one could be a 'c' — this would indicate a computer dealer, E would indicate an electrical dealer and so on. As I have said 10 spaces are available so space 1= type, space 2= good/bad payer, space 3 whether local or not and so on. These characters are entirely left to your choice and do make for a very useful way of printing selective lists.

Labels

You can format your label easily and specify which fields you want printed. Also, you can print customer lists, telephone lists and so on purely by specifying the field to be output.

Early one

The programming on this is not up to the standard of stock control and database and I suspect that it may be an early version. I do hope that, like the database, it will be re-written as it has a lot of potential and could be a very useful aid to anybody who has a small business as well as to home users. Comments regarding capacity are the same as with the previous programs.

Gemini software

When I first came across Gemini software it was a disaster. Ideas were good but everything was let down by abysmal programming and poorly thought out design. It looks as if they are coming on leaps and bounds as their later offerings are well worth a look. If Gemini can update some of their earlier programs I am sure they will find more people buying their reasonably priced stand alone modules.

MASTERING MACHINE CODE

To AND or to OR, this is one of the questions posed by A.P. and D.J. Stephenson in their examination of logical operations.

THE WORD 'LOGIC' IS USED in a variety of ways. It is normally used, in a loose sense, to indicate clarity of thought, particularly the means by which conclusions are drawn by careful analysis of facts. The art of 'correct' thinking was pioneered by Aristotle who founded a school of thought which subsequently became known as Aristotlean Logic. It was ponderous in form and, because it was based on common language, was of little practical use apart from the intellectual prestige which its devotees attracted.

It was left to the 19th century Irish schoolmaster, George Boole, to sort things out. He extracted the important ideas of Aristotle from the mass of semantic nonsense which had grown round them. In effect, he changed logic from an art into a respected branch of pure mathematics when he published a relatively small book entitled, 'An Investigation into the Laws of Thought'.

Although Boole's ideas made little impact at the time, Claude Shannon (a pioneer of Information Theory) and later John Von Neuman (the father of the modern digital computer) realising its value in the analysis of complex switching circuitry, made valuable contributions to the subject, including the introduction of a new, and easier to understand, set of symbols.

Logic, as far as we are concerned here, is really the study of the various switching actions which take place within silicon chips and how such actions can be simulated by software. We should remember that even a microprocessor itself is little more than a complex arrangement of *switches* or, as they are more rightly called, *logic gates.*

Logic gates

Those whose interests extend to both hardware and software will probably agree with the following simple definition:
A logic gate has one output and one or more inputs. The logic state of the output depends on the logic states applied to the inputs.
By the term 'logic state' we mean a '1' or a '0'. Although most readers may not be too interested in the electrical details, it is worth mentioning that, as far as the 6510A is concerned.
A voltage around 3 to 5 volts is recognised as 'logic 1'
A voltage lower than about one volt is recognised as 'logic 0'.

There are several types of logic gate but only the following three are of particular interest to the machine code programmer:

The AND gate
Output is at logic 1 only if all inputs are at logic 0.

The INCLUSIVE-OR gate
Output is at logic 1 if at least one of the inputs is at logic 1.

The EXCLUSIVE-OR gate
Output is at logic 1 only if the two inputs have different states.

The accepted symbols for these gates are shown in Figure 6.1

There are three instructions in the 6510A which simulate gate action. The mnemonics codes and addressing modes available follow:

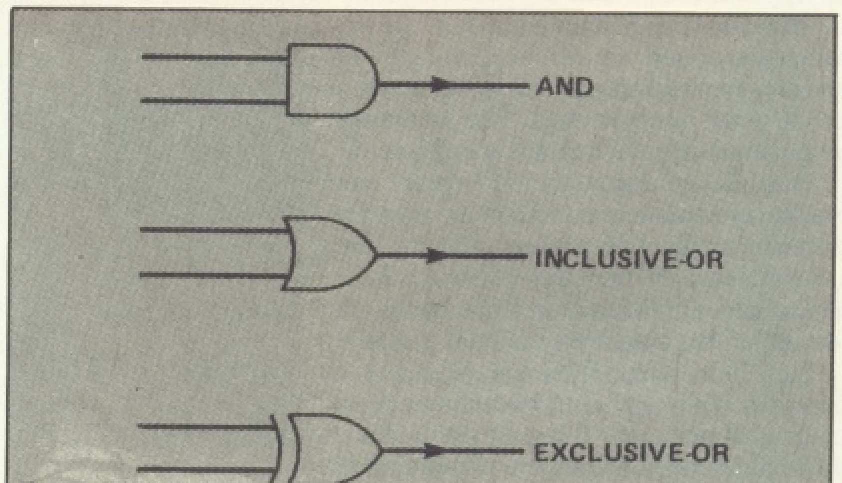


Figure 6.1

THE AND instruction

Assembler	Hex code
AND # $xx	29 xx
AND $xx	25 xx
AND $xxxx	2D xx xx
AND $xx,X	35 xx
AND $xxxx,X	3D xx xx
AND $xxxx,Y	39 xx xx
AND $(xx,X)	21 xx
AND $(xx),Y	31 xx

The ORA instruction

Assembler	Hex code
ORA # $8xx	09 xx
ORA $xx	05 xx
ORA $xxxx	OD xx xx
ORA $xx,X	15 xx
ORA $xxxx,X	ID xx xx
ORA $xxxx,Y	19 xx xx
ORA $(xx,X)	01 xx
ORA $(xx),Y	11 xx

The EOR instruction

Assembler	Hex code
EOR # $xx	49 xx
EOR $xx	45 xx
EOR $xxxx	4D xx xx
EOR $xx,X	55 xx
EOR $xxxx,X	5D xx xx
EOR $xxxx,Y	59 xx xx
EOR $(xx,X)	41 xx
EOR $(xx),Y	51 xx

What use are they?

It is all very comforting to know that these logical instructions are available but the most obvious question readers will ask is — what use are they? Well, there will be times, particularly if interests extend to the *control* or peripheral gadgets, when you may need to opertate on *particular bits* within a byte rather than on the entire byte. For example, we may wish to ensure that bit 3 in the byte is set to 1 *without altering the remaining bits.* As another example, we may wish to clear bits 3, 5 and 7 but set bit 2. These operations fall into one of three main categories:
(a) Clearing selected bits in a byte to '0' without disturbing the other bits. The AND instruction is involved.
(b) Setting selected bits in a byte to '1' without affecting the other bits. The ORA instruction is involved.
(c) Changing selected bits in a byte from their present to their opposite state without affecting the other bits. The EOR Instruction is involved.

The mask pattern

Knowing which instruction to use, out of the three possibilities, is only half the battle because there still remains the problem of working out the correct bit pattern for the operand — called the mask. Think of it in the following way:

1. Each bit in the mask, and its corresponding bit in the accumulator, form the two INPUTS of a logic gate.
2. After the instruction is performed, the accumulator bit is the OUTPUT of the gate.

As the table above showed, the logic instructions can be used with a variety of addressing modes but we shall use only the immediate mode for illustration. It is necessary to remind readers that the bits, within a byte, are always numbered bit 0 to bit 7, the least significant bit at the right being bit 0.

To clear selected bits to 0

Use AND with an operand mask designed as follows:
'1's in the mask will leave corresponding bits in the accumulator unchanged but '0's in the mask will ensure corresponding bits will remain at, or be reset, to 0.
Example: To ensure bit 5 in accumulator is set to '0', use:

AND # $DF

To see why, remember that an AND gate requires both inputs to be 1 in order for the output to be 1. Examine the following accumulator example:

Accumulator before # $DF	0111	1001
Mask pattern $DF	1101	1111
Accumulator afterwards	0101	1001

Note carefully that the accumulator is left exactly the same as before except that bit 5 is now 0 instead of 1.

To set selected bits to 1

Use ORA with an operand mask designed as follows:
'0's in the mask will leave corresponding bits unchanged but '1's in the mask will ensure corresponding bits are set to '1'.
Example: To ensure bits 2 and 6 in the accumulator are set to '1' use: ORA # $44

To see why, remember that only one of the inputs to an inclusive - or gate need be 1 in order for the output to be 1.

Accumulator before ORA	$44	0011	0101
Mask pattern $44		0100	0100
Accumulator afterwards		0111	0101

Examine the above accumulator example:

Note bit 6 has been changed from 0 to 1 but bit 2 happened to be at 1 anyway.

To change selected bits

Use EOR with an operand mask designed as follows:
'0's in the mask will leave corresponding bits unchanged but '1's in the mask will ensure corresponding bits are changed.
Example: To ensure bits 3, 4 and 5 in the accumulator are changed, use:

EOR # $38

Remember than an exclusive-or gate gives an output at 1 only if the inputs differ.

Accumulator before EOR # $38	1011	1100
Mask patter $38	0011	1000
Accumulator afterwards	1000	0100

Examine the above accumulator example.

The novelty behind the following snippet of useless (?) knowledge might intrigue some readers:

Exclusive-oring data with itself always results in zero.

Accumulator before EOR #$9D	1001	1101
Mask pattern $9D	1001	1101
Accumulator afterwards	0000	0000

For example, if A contains $9D and we write EOR # $9D, the result in the accumulator is zero as we can see above.

Non-arithmetic logic

Logic operations have no connection whatsoever with the normal laws of arithmetic because, for one thing, there is no carry action. Each bit is an individual entity and quite contemptuous of the feelings of neighbouring bits. To see the absurdity of trying to equate logic results with arithmetic results, consider the result if we AND 2 and 3 together — instead of ADDING them:

2	0010	0010
3	0000	0011
Result of ANDing	0000	0010

This means that 2 AND 3 = 2!

Logic and input/output ports.

Some computers already have a socket at the back marked 'User Port' or they have facilities for including one. These are used for connecting digital operated devices such as the points of model railways, cranes, garage doors, intruder alarms, robots, special lighting effects, etc. Eight wires and a couple of control lines can be connected to the output port.

The state of each line, and therefore the on/off state of the devices can be controlled by storing data patterns in an output port register. This is an area where the three logic instructions can be used most effectively because of the necessity to control the state of individual bits without affecting the others.

One's complement of accumulator

It is sometimes appropriate to change all the bits within a byte. That is to say, change all '1's and all '0's to '1'. This is sometimes called 'flipping' the bits and the result is known as the 'one's complement' (refer back to Part 1 of the series.) Assuming the data is already in the accumulator, it is clear from the above treatment that flipping all the bits can be achieved by using:

EOR # &FF

Two's complement of accumulator

The two's complement of a number is really the one's complement with an extra 1 added. Unfortunately, we can't add the extra 1 by incrementing because the result is in the accumulator and you will remember that no direct incrementing instruction exists for this register. A possible coding is then:

```
EOR # &FF
CLC
ADC # 1
```

An alternative method is to rely on subtracting A from zero. The 6502, and nearly all other microprocessors, use two's complement arithmetic for addition and subtraction. It follows that by subtracting a number from zero, we obtain the two's complement because 0 – X = –X. So to obtain the two's complement of the accumulator, we must first store the contents in a memory location. Then after clearing the accumulator, the original data can be subtracted from the accumulator by use of SBC.

Finding state of a particular bit

It is sometimes important to find out the state of one particular bit within a byte. This can be done by first loading the byte into the accumulator. All the bits, except the one of interest, are then cleared to zero by using an AND mask. If the result is then tested by BNE or BEQ, a zero result proves that the bit of interest was indeed a '0' and a non-zero result proves that it was a 1.

An alternative, and simpler method, can be used if the bit of interest happens to be in bit

6 or bit 7 position because the BIT instruction caters specifically for testing these two positions. Suppose, for example, we write BIT &2000. This causes the state of bits 6 and 7 at this address to be copied into the V and N positions in the Status Register respectively. The original state of bit 7 can then be tested by using a BMI branch (which tests N) or bit 6 by a BVS branch (which tests V). There is, however, another operation which takes place during the BIT test, which can be either a nuisance or a bonus depending on the circumstances. The contents of the operand address are logically ANDed into the accumulator. If the accumulator holds valuable data at the time of the BIT test, it is important to store the original contents first.

Shift instructions

Assembly	Hex code
ASL A	OA 06 xx
ASL $xx	OE xx xx
ASL $xxxx	
ASL $xx,X	16 xx
ASL $xxxx,X	1E xx xx
LSR A	4A
LSR $xx	46 xx
LSR $xxxx	4E xx xx
LSR $xx,X	56 xx
LSR $xxxx,X	5E xxxx
Rotate instructions	
ROL A	2A
ROL $xx	26 xx
ROL $xxxx	2E xx xx
ROL $xx,X	36 xx
ROL $xxxx,X	3E xx xx
ROR A	6A
ROR $xx	66 xx
ROR $xxxx	6E xx xx
ROR $xx,X	76 xx
ROR $xxxx,X	7E xx xx

The BIT test

Assembly	Hex code
BIT $xx	24 xx
BIT $xxxx	2C xx xx

The assembly and hex code form of the BIT test are as above.

The Shift and Rotate instructions

To shift a register or memory byte means to push the bit pattern sideways by one bit position either to the left or to the right. The coding details of the two instructions which produce shift action ASL (Arithmetic Shift Left) and LSR (Logical Shift Right) are shown below. Rotating a register or memory byte is similar to shift action except bits, which would normally overspill at the end are re-inserted again at the other end. The two instructions are ROL (ROtate Left) and ROR (ROtate Right).

The shift and rotate instructions are unique in that one of the available addressing modes is Accumulator Addressing so they can act directly on the accumulator or they can act on memory locations. If the action is required on the accumulator, the mnemonic op-code must be followed by A. Note that an operand byte is not required. For example, ASL A will shift the contents of the accumulator one place left. A common mistake, when using an assembler, is to just write ASL and forget to follow it with A. This would be unrecognisable code. The instructions must either have an A or an operand address following the mnemonic. If the hex code is entered directly without the use of an assembler, the above warning does not apply because the hex code itself distinguishes between accumulator or memory addressing.

Note that in all four instructions, the C bit is involved and can be thought of as the 'ninth bit'. LSR and ASL provide essentially 'open-loop' actions because bits can drop out or be lost if the C bit is already occupied. On the other hand, ROR and ROL provide 'closed-loop' actions because if any bit is pushed out at one end it is re-inserted at the other. It is easier to follow the action of these four instructions by means of simple diagrams as shown in Figure 6.2

Figure 6.2



Although the C bit appears to be joined to the registers, we should bear in mind that it is physically located up in the status register of the microprocessor.

Single byte multiplication

Subject to overspill into the carry, shifting left by using ASL will multiply by two each time, so four consecutive ASL operations will multiply existing data by sixteen. It must be understood that simple shift or rotate instructions can only multiply by an integral power of two. If, for instance, we want to multiply by 5, we must shift the accumulator left twice and then add the accumulator to itself once.

Single byte division

Division by two is achieved by LSR although we must remember that the overspill from the right (from the lsb) goes into the carry. As a matter of interest, the reason why LSR is named Logical Shift Right is due to this very reason. It is arithmetically absurd for carry status to be in the lsb position, hence it is deemed to be 'logical' rather than 'arithmetical' in nature. This is in contrast to ASL (Arithmetic Shift Left) where the carry action is natural because it is positioned at the msb end. Unless the programmer is sure, perhaps by prior local knowledge of the data limits, multiplication and division techniques rely heavily on careful checking of the carry status. In double length working, the carry bit provides a continuity link between the low and high bytes of the composite number.

Double-byte multiplication

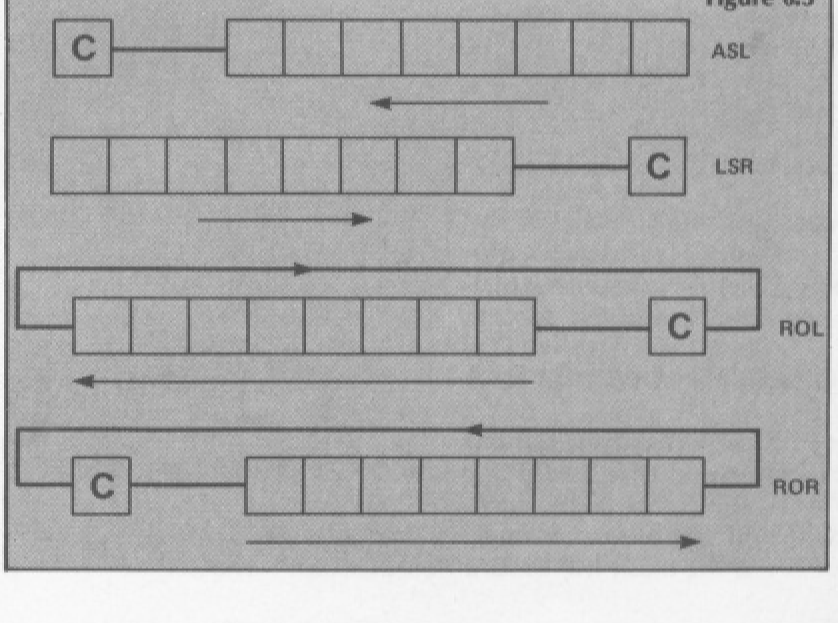
This provides a useful exercise in shift and rotate instructions. Although two separate locations are used for each double byte number, the C bit provides continuity between the two. Although ASL and ROL both multiply by 2, the carry can be a problem if they are not chosen wisely. No carry must be allowed to enter the lower order byte from the right so ASL is appropriate. On the other hand, the higher order byte must take into consideration the carry from the right so ROL must be used. Assuming the data is in two bytes of memory, the coding would be:

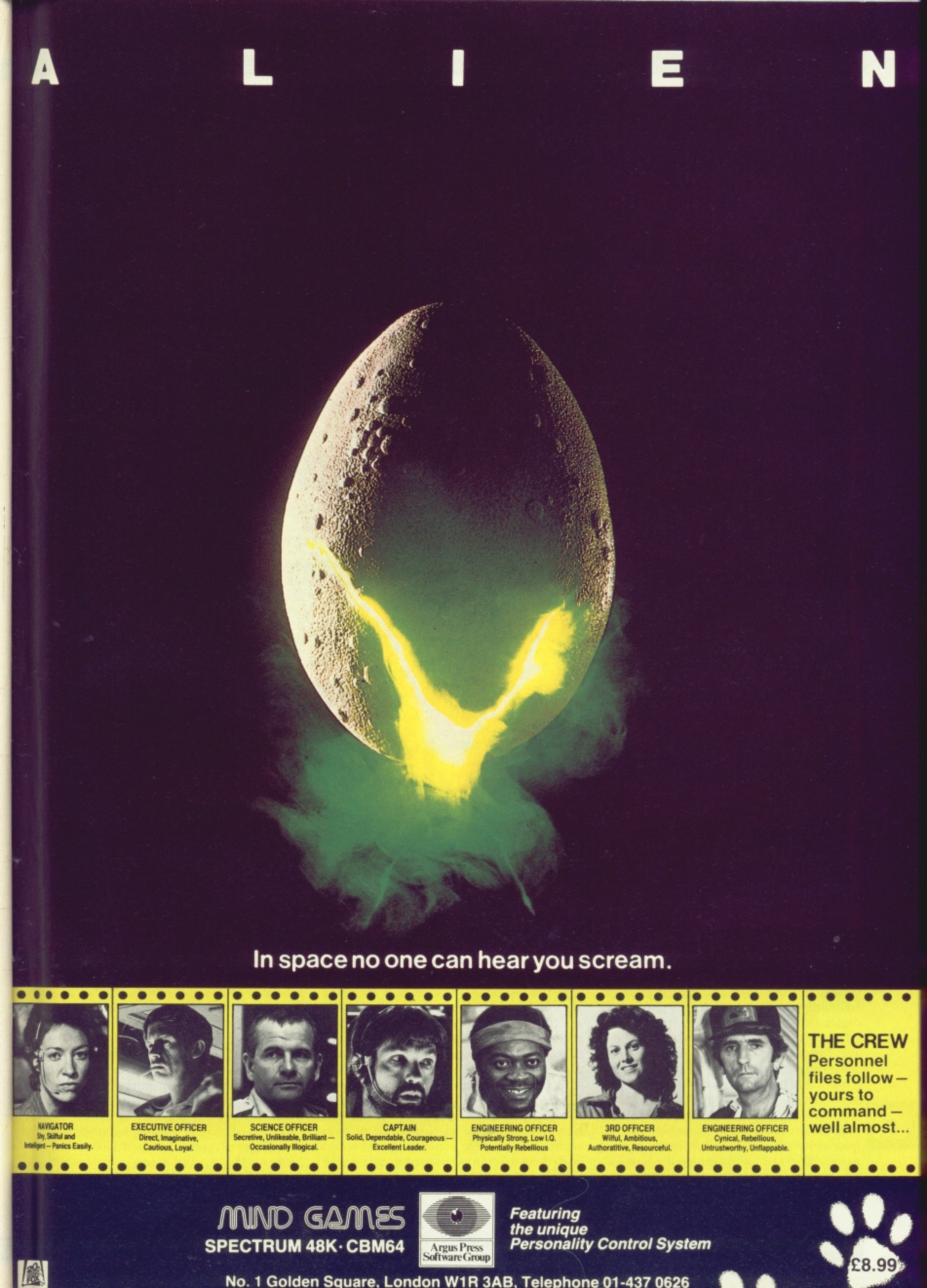
```
ASL  low byte
ROL  high byte
```

Double byte division

Division is virtually the opposite to multiplication so the higher order byte must be attacked first and a carry must not be allowed to enter from the left. This suggests LSR is correct for the first step. The lower order byte must receive a carry (if any) from the left so the correct instruction is ROR. Assuming the data is in two bytes of memory, the coding becomes :

Figure 6.3





A L I E N
In space no one can hear you scream.
NAVIGATOR
Shy, Skilful and Inteligent — Panics Easily.
EXECUTIVE OFFICER
Direct, Imaginative, Cautious, Loyal.
SCIENCE OFFICER
Secretive, Unlikeable, Brilliant — Occasionally Illogical.
CAPTAIN
Solid, Dependable, Courageous — Excellent Leader.
ENGINEERING OFFICER
Physically Strong, Low I.Q. Potentially Rebellious
3RD OFFICER
Wilful, Ambitious, Authoratitive, Resourceful.
ENGINEERING OFFICER
Cynical, Rebellious, Untrustworthy, Unflappable.
THE CREW
Personnel files follow — yours to command — well almost...
MIND GAMES
SPECTRUM 48K · CBM64
Argus Press Software Group
Featuring the unique Personality Control System
£8.99
No. 1 Golden Square, London W1R 3AB, Telephone 01-437 0626

STUCK IN THE MUD

Richard Bartle immerses himself in MUD. Follow his footsteps into the Jungle.



COMMODORE OWNERS WHO KEEP themselves abreast of happenings out there in the big, bad, computer world, won't have failed to notice the new network which has been set up especially for CBM64 owners, Compunet. They'll also be aware that while it's quite a promising system, it's still in its infancy and hasn't too many games available on its pages.

This should change fairly quickly, because there's an area of the network known as The Jungle, where users can upload their *own* pages, including their own software, and even make people pay if they want to play it! Most of these will be games specially designed for the 64, which will download into your machine and use the modem as a dongle to stop you giving it to other people (or, even worse, selling it!). There's one program, however, which doesn't do that; you never get a copy of it zapped down the line at you because it runs on whacking great big mainframe computers, the same ones which the Compunet system itself uses. It uses more disc space than you can store on a floppy, never mind a cassette, and (not surprisingly) it's the only game of it's kind in the world. This program is called MUD, an acronym for Multi-User Dungeon. It's a normal adventure game in virtually every respect except one: you don't play alone.

Multi-user dungeon

MUD is the first adventure game where more than one person can play at the same time. To understand the full impact which this has on the game, you really have to play it. The difference made by the fact that other people are in there with you is so profound that it's very hard to get over in an article such as this. It's just such an incredible extra dimension that it leaves ordinary games standing. With other players around to thwart your ambitions, or help you when you're down, to chat with you (while perhaps relieving you of your belongings!), MUD improves on the basic concept of an adventure game by such an extent that it just *has* to be the way computer games are going to go in the future. MUD on Compunet may be the only commercial version available for the moment, but within a couple of years there will be multi-user adventures sprouting up all over the place. The whole computer games market may never be the same.

If MUD's such a good idea, then, why hasn't it been thought of before? Good question! The problem is that in order to manage such a piece of software you need very powerful computers. Micros just aren't up to it. No-one is going to buy a mainframe computer with half a million pounds just to see if they can write a multi-user adventure game! Also, it's only recently that the micro boom has started to give way to the communications boom, with modem sales rising as micro sales start to drop. Up until now, there's been hardly any market for games which you can just play over the phone lines. Now, however, the growing number of modem owners looking for something new to do with their machines has prompted people like Compunet to set up networks to tap the market.

Advent

In order to trace the development of MUD, we have to go back to 1979 at Essex University. There, undergraduates used to spend their free time on the University's mainframe computer playing this new game they'd discovered. They knew it as Advent, but these days it's called Adventure or Collosal Cave. Judging by the impact it has had on the world of computer games, perhaps the name Advent is more appropriate!

One of those undergraduates, Roy Trubshaw, played Advent and liked what he saw. There were a few things which really niggled him, though, for example, the poor command parser (verb-object

pairs only). He was also annoyed by the fact that Advent was a one-off, and if he wanted to make the program work for another fantasy world it would have to be done from scratch. Why bother rewriting all those routines to move, drop objects, kill monsters and the like when most of them are common to all adventure games? What he envisaged was a game which had its own built-in adventure-designing language, so you only had to say a few things and it *knew* what to do with them. If all adventures have tables of rooms, objects, room connections and the like, what is to prevent your making them data instead of part of the program? And take our game-dependant stuff too; like having it check there's a bear following you every time you go round the command loop so it can inform you you're being followed by a bear..?

The other major disadvantage he saw in Advent was that it was only a single-user game. No-one else could be in there with you to help you out in times of trouble, or give you times of trouble if you had more treasure than they did! Surely a game along those lines would be much more fun?

And so he set about writing such a game — called MUD. It had a language of its own to define the world, and because Essex University's powerful (by the standards of those days!) DEC-10 computer did timesharing, it wasn't too difficult to arrange it so several people could play at a time. The thought of what would happen in the future if everyone had a computer of their own which they could connect to a network to play games of this kind, just didn't concern him; he was doing it solely out of interest and love of programming.

A helping hand

What Roy came up with was a bare-bones system, which had a programmable world, a passable language parser, and multi-user capabilities. Now one of Roy's friends was a chap by the name of Richard Bartle. I'd say he was an expert games-player and a programmer of the most elite class, except since he's me you'd think I was boasting! In spring 1980, Roy had gone about as far as he wanted to with MUD. I'd helped him with ideas from the start, but the programming was all his own work. However, Roy's great love is writing programs, and he's not particularly interested in designing adventures, so I gradually took more of a part in designing the game, starting with adding new rooms to the world which it modelled and gradually moving over to adding bits to the code. When Roy left at the end of his 3rd year, I took the game over and have never looked back!

The first thing I did was to rationalise some of Roy's experiments. The multi-user aspect hadn't been explored in full, and there were anomalies (such as if two people were in an underground room and one had a torch, the other couldn't see). I fixed those sort of things, and added in a few more interactive commands like stealing, helping, giving. I increased the number of rooms gradually to its present number of 418, and put in an appropriate number of new objects (an easy thing to do since we had the Multi-User Dungeon Definition Language — MUDDLE!). What the game didn't have was a purpose, however, so I put in the concept of scoring for treasure, and having levels of experience based on the amount of treasure you'd accumulted in previous games.

In order to debug MUD when we'd just stuck in new rooms, we'd always had a "debug mode", or "wizard mode" as we used to call it. If a new room complex had been added, then to test it out we might normally have to get an axe, chop down a tree, fetch a light source, and go beneath the tree to explore the new rooms. Wizards could fly to any room, and they glowed in the dark.

Snoop

About this time, I had a spare afternoon and decided to put in a new feature, the "snoop". With this, one player (if they were in wizard mode) could sit and watch what was on the screen of another player, without that other player knowing. The original intention was so you could see common mistakes people made, and try to get the game to cope with them. It turned out to be far more useful than that!

When I put in snoop, I spent the next 3 hours enraptured by watching other people stumble about the game and make complete fools of themselves! It was tremendous fun! The time just flew by, and I resolved that I'd better make this facility more generally available. So, when people got a certain number of points for playing, they were given the password to wizard mode and obtained the same powers as I had.

Wizard mode works really well. Non-wizards (mortals) all the time witness the power of wizards, and strive to make it themselves. To date, 52 players out of maybe 3 or 4 thousand who have tried the game have managed to make it to the top. We also have female wizards, who are called witches, so there's a generic term, wiz, to mean both wizards and witches. Wiz's still play if they can, too, because the game is never ending. When you're a wiz, there are still fresh supplies of new people coming in to watch as they progress through the game, and you have plenty of friends in there anyway if you just want a chat. Although wiz's are able to do immensely powerful things (there's a CRASH command — and it makes MUD do just what it says!), they rarely do. This is because they've been mortals themselves and know how heart-breaking it is for someone to interfere with the game and make them lose all their points. They tease mortals, yes, but always reward them with a few treasures afterwards to show they're really nice deep down...

The rest of the world found out about MUD from ex-Essex players who left with a yearning to hack and slay in the world of MUD. The grapevine was the only way people heard about the game for ages afterwards, until the present flurry of articles in the Computer press. Now I get 5 or 6 letters a day from people asking how to access MUD.

Playing the game

So how can you play the game? Well, there are currently two 'open' versions in Britain (and one in Norway!), one of which is free and one of which isn't. The free one is based at Essex University, and is the original. Because of this, it gets changed whenever I feel like it and is prone to crashes (OK, so it crashes at least once a night!). Also, it keeps extraordinary hours, like midnight to 7am, or, if the computer is exceptionally busy, from 1am to 7am. The University may not charge money to play it, which is very decent of them and makes you glad you pay your taxes, but BT do charge money, and to access MUD via PSS (the BY national network) costs at least £2.50 an hour. The second MUD site is Compunet, which comes to around £3.50 an hour, but doesn't crash so often and has more civilised hours.

Compunet will be sole distributors of the current version of MUD for some time, but work is already proceeding apace at the new, improved version! There has to be a new version, because now people know how good MUDs are they'll start designing their own, and we'll have a whole bunch of them appearing before you know it. The best these will be able to do is imitate MUD, however, whereas with the order of 25,000 hours playing time behind it, the Essex MUD has lots of experience which can be drawn upon in creating an even better version (if such a thing is conceivable!) (oops, did I just boast?).

For the moment, though, MUD remains unique. So if Father Christmas brings you a Commodore modem for Christmas, and you find yourself huddled over a micro on December 25th, snow falling outside, a mug of hot soup beside you, as you tap through gloved fingers at a keyboard that's beginning to freeze over, remember it's only your body that's feeling the cold. The *real* you is perhaps hundreds of miles away in MUD, sword in hand, wand at the ready, doing battle with who knows what and who knows whom, to force your way against the odds up to wiz. MUD is always warm, of course — it *is* in the Jungle, after all!

FIRST AID



for your 64

Computers, like people, are fallible. They need the right combination of code and care to perform effectively in the business or the home. And that requires first hand knowledge from you to create a healthy operating environment for your Commodore 64.

Knowledge about machine language, about the lesser known qualities of the 64, about the disc drives, graphics, and about the tricks and tips to keep your 64 on line. That's why First Publishing has now launched in the UK a series of high quality books and software packages to provide a complete health care kit for your 64. Commodore 64 users throughout Europe have already found it a tonic. We think you will, too.

For a brochure on all the Commodore 64 books and software packages available from First Publishing, please fill in the coupon and send to: Amanda York, First Publishing, Unit 20B, Horseshoe Road, Horseshoe Park, Pangbourne, Berkshire.

Name ____________________

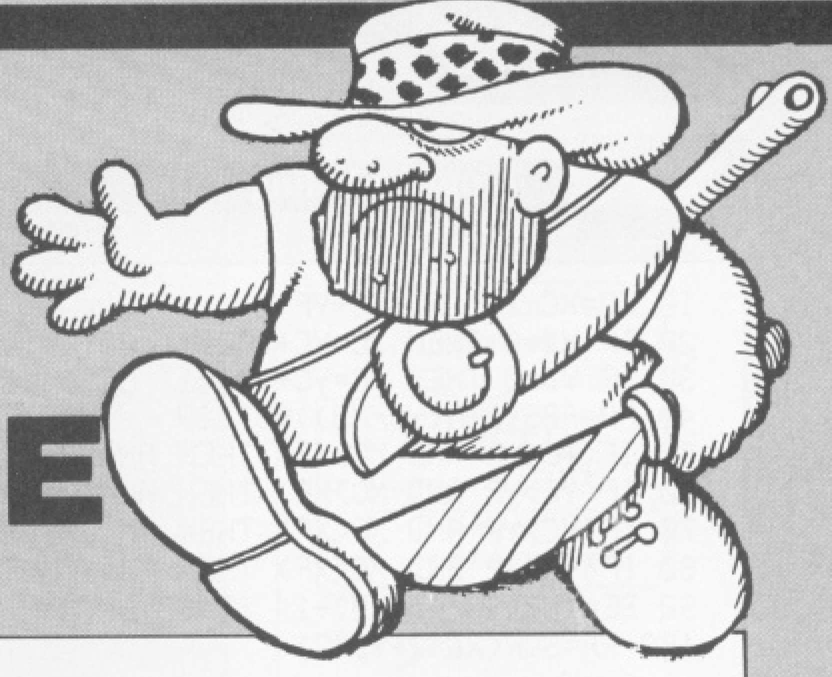
Address ____________________

Or ring *Amanda York* at (07357) 5244 or Calco Software at 01-546 7256 ycom/3

1st

FIRST PUBLISHING LTD

SETTING OUT ON AN ADVENTURE



If you want to try your hand at writing adventures but don't know where to start, let Allan Webb show you the way into this complex subject.

ONE OF THE MOST GRIPPING ASPECTS of adventures is that writing them is as challenging and enjoyable as playing them. The programming aspects tied up in adventures are various, including artificial intelligence, data compression and graphics. In this series of articles, I intend to discuss some of the aspects of writing adventures. I don't intend to spoon feed all the code necessary for you to write a complete adventure; there are enough books on the market which do that job. Instead, I want to give a collection of ideas and routines which, I hope will trigger your own ideas and perhaps give your games something extra. Owing to space limitations, the number of listings will be limited to a few machine code utilities.

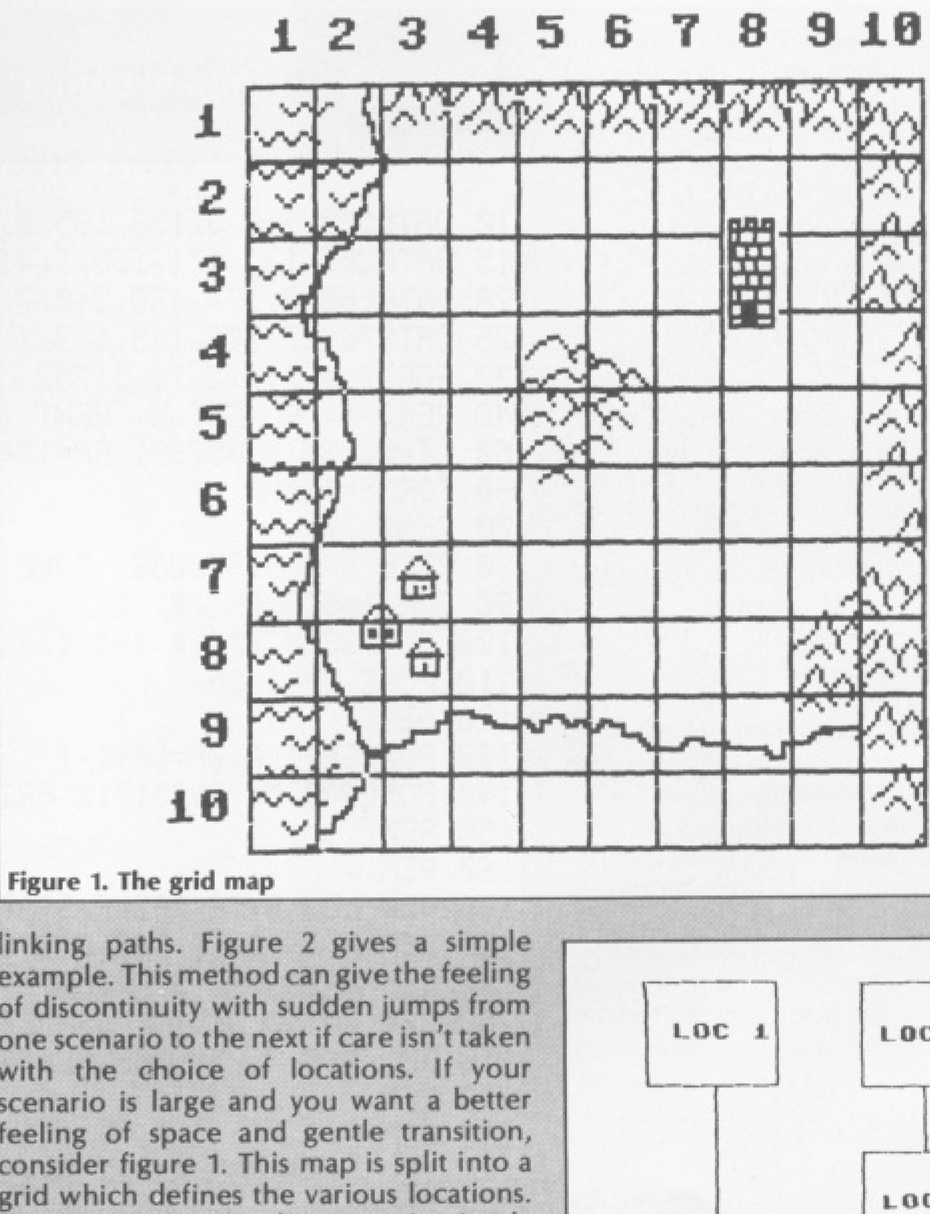


Figure 1. The grid map

Plotting

The most crucial phase of any adventure is the writing of the scenario and plot. It's the quality of the plot which will make or break your masterpiece. Before doing anything, I suggest that you look at as many other adventures as you can, note what they do and ensure you don't copy them. Nobody likes a copy whilst everyone will admire an original. If you must research for ideas try the written word. You should decide in detail what happens where and who is involved.Don't be tempted to start coding until you're happy with the plot.

Mapping

There are two general ways of mapping adventures, each with its own particular feel. The first type of map uses discrete locations, each with a description, with linking paths. Figure 2 gives a simple example. This method can give the feeling of discontinuity with sudden jumps from one scenario to the next if care isn't taken with the choice of locations. If your scenario is large and you want a better feeling of space and gentle transition, consider figure 1. This map is split into a grid which defines the various locations. Your position on the map is simply defined by a pair of coordinates. Despite the larger number of locations, you don't necessarily require a larger number of descriptions. For example, all areas of mountains or all areas of open land will have the same descriptions, so that perhaps half a dozen descriptions can cover a large percentage of the locations.

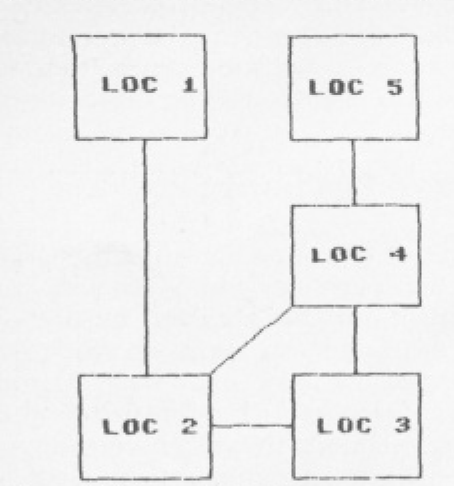


Figure 2. Map with paths

The feel and atmosphere of your game will depend on the quality of the descriptions of the scenario. Both graphics and text can be used to provide descriptions. There is a lot of silly snobbishness when it comes to whether graphics or text should be used. In my opinion graphics are rather too RAM hungry and you get too poor a return from them to justify their use. In fact, good use of text can give an excellent atmosphere; try the Level 9 or Infocom

Program Listing 1

```
10 XD=XC-XP: YD=YC-YP
20 IF XD=0 THEN XC=XC+.0001: GOTO 10
30 IF YD=0 THEN YC=YC+.0001: GOTO 10
40 PA=ABS(ATN(YD/XD))*57.29
50 IF(YC<YP AND XC>XP) THEN AN=90-PA
60 IF(YC>YP AND XC>XP) THEN AN=90+PA
70 IF(YC>YP AND XC<XP) THEN AN=270-PA
80 IF(YC<YP AND XC<XP) THEN AN=270+PA
90 BE=INT(AN/45+.5)+1: IF BE=9 THEN BE=1
100 RA=SQR(XD↑2+YD↑2)
```

games and see. I would simply add that you don't find pictures in most decent novels.

In this part, I will deal with text, but fear not, I will discuss graphics in a later section. You can split the description of a location into three sections. First there is the main description. You know the sort of thing..."You are in a long room filled with stone emporer penguins". This section never changes during the game. There then follows a variable section describing fixed items which might change status. For example..."There is an open door, the light is on." to give real variation, a third section can be included. This will describe "one off" occurances such as "A herd of hippos is walking by" or "An old man is wrestling with an Aardvark in the corner."

Where the large scale map approach is adopted, the use of the third section is vital to prevent the scenario becoming monotonous. You can also include other variables to add spice. Everyone knows the boring old situation where you enter a cave and must have a torch because it's dark. This can obviously be extended to cover other areas. For example why not have day and night periods, or how about including the weather. The occasional snow storm or monsoon can be used to make the game more difficult or hazardous.

Relative Positions

Because of the open nature of the large map approach, a better feeling of movement can be obtained by use of relative positioning. Imagine you are standing to the south of a village. As you move northeast, your position and distance relative to the village will change. If the village's position on the map is known, the exercise is trivial. Listing 1 shows how to calculate your bearing BE and range in arbitrary units (RA) from a point XC,YC. Your position is XP,YP. BE will have a value between 1 and 8 can be used to print the bearing as a point of the compass such that north=1, northeast=2 etc.

This concept can also be applied to major landmarks such as mountains, deserts and the sea.
You can then print after your description something like:

"4 leagues to north lies a castle."

Text storage

OK, enough theory, now some harsh reality. Where on earth can all this marvellous text be stowed. Text tends to be memory hungry. If, for example, you have fifty locations each with a three line description, you will loose 6K of RAM. The simple method of saving the text is by use of DATA statements and strings. All very nasty and wasteful. How about sequential files on cassette or disk? Again not ideal. The most elegant approach is to use the spare RAM behind the ROMs. Specifically, the 8K behind the BASIC ROM. Listing 2 gives a routine for printing text stored in RAM.

Program Listing 2

```
10 DATA172,132,3,136,185,0,206,133,170,185,0
15 DATA207,133,171,165,1,41,254,133,1
20 DATA160,0,177,170,240,7,32,210,255,200
25 DATA76,22,202,165,1,9,1,133,1,96,0
30 FOR I = 51712 TO 51752
40 READ X: POKE I,X: NEXT
50 LT=52735:HT=52991:SP=10*4096:MN=0:NM=3
60 FORMN=1TONM
70 READ I$
80 POKE HT+MN,SP/256: POKE LT+MN,SP-INT(SP/256)*256
90 FOR I=0TOLEN(I$)-1
100 CH=ASC(MID$(I$,I+1,1))
110 POKE SP+I,CH
120 NEXT
130 POKESP+I,0:SP=SP+I+1
140 POKE900,MN:SYS51712:PRINT:NEXT
150 REM
160 REM
170 REM LOW BYTE TABLE STARTS AT 52735
180 REM HIGH BYTE TABLE STARTS AT 52991
190 REM MESSAGES START AT 40960
200 REM NM HOLDS THE NUMBER OF MESSAGES
210 REM 900 HOLDS THE NUMBER OF THE MESSAGE TO BE PRINTED
220 REM
230 REM
240 REM MESSAGES START HERE
250 REM
260 REM
30000 DATA "MESSAGE 1"
30010 DATA "MESSAGE 2"
30020 DATA "MESSAGE 3"
```



This routine both saves and recalls the text from the block of RAM starting at SP. The start of each message is kept in LT and HT. If you want to use the routine as a data loader simply change line 140 so that holds just NEXT. The following simple line will print message number MN at the

current cursor position in the current colour:

10 POKE 900,MN: SYS 51712

Lines 10 to 40 must be included in your program and executed before you try to print a message. You can use this routine with any section of RAM below $C000 (49152).

I've ignored the area behind the Kernal ROM since it can be used for high resolution graphics. Listing 3, however, will enable you to use this area for storage if you wish:

you switch out the ROM before saving and switch it back in afterwards; both from within the monitor. Saving data from behind the Kernal ROM is tougher since the input/output chips are in that area. A loader such as listing 2 is the simplest way of loading data into the area above $CFFF.

Layouts

Finally, I want to say something about the layout of the complete adventure. If you're using BASIC, you must take care with the layout to ensure maximum

Program Listing 3

```
1 DATA76,134,202,76,169,202,32,212,202
2 DATA165,20,133,158,165,21,133,159,160,0
3 DATA165,1,141,231.3,41,248,120,133,1
4 DATA177,158,141,232,3,173,231,3,133,1,88
5 DATA96,32,212,202,165,20,133,158,165
6 DATA21,133,159,32,212,202,165,20,141,232
7 DATA3,160,0,165,1,141,231,3,41,248,120
8 DATA133,1,173,232,3,145,158,173,231,3
9 DATA133,1,88,96,32,253,174,32
10 DATA138,173,32,247,183,96
15 FOR I = 51840 TO 51933
20 READ X: POKE I,X: NEXT
10000 A$="THIS IS BEHIND THE KERNAL ROM":SA=13*4096
10010 FORI=1TOLEN(A$)
10020 CH=ASC(MID$(A$,I,1))
10030 SYS 51843,SA+I,CH:NEXT
10040 FORI=1TOLEN(A$)
10050 SYS 51840,SA+I
10060 PRINTCHR$(PEEK(1000));:NEXT
20000 REM
20010 REM
20020 REM POKEALL....PUT VALUE IN LOCATION 1000
20030 REM SYS 51843, ADDRESS, VALUE
20040 REM
20050 REM PEEKALL....SYS 51840, ADDRESS
20060 REM VALUE IN 1000
20070 REM
```



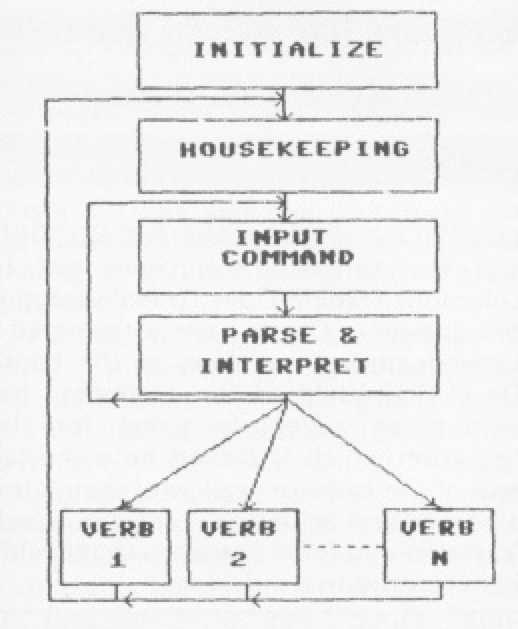
This routine enables you to POKE and PEEK any section of RAM. Lines 10000 to the end show how to store text by POKEing it into RAM. It's slow but works. Lines 1 to 20 must be included in your program if you want to use the routine.

A quick look at these routines will show that a fair amount of data is associated with any adventure. I would recommend any beginners to use a data loading program to put all the data into RAM before loading the main adventure. More advanced souls can save the data using a machine code monitor. Data which hides behind the BASIC ROM can be saved using a monitor provided

speed. Figure 3 shows the algorithm I use. As you can see, it's essentially linear. The first box involves setting up variables, loading machine code and other sundries. The second section performs the repetitive steps such as updating the display, checking for night, checking to see if you've won, looking after the movement of characters etc. This section is executed after every "turn".

The next two boxes take your command and interpret it. You are warned of nonsense commands at this stage. If your command is valid the final step performs the command. I favour the use of ON GOSUB or ON GOTO

FIGURE 3.



constructs to "fan out" the flow, so that parallel rather than serial routines are used. Again this increases speed.

Each verb routine has a similar form:

Validity check
:
:
Perform action
:
:
Catchall

The validity check makes sure that the object is present or is carried and inputs a suitable command and exits if it isn't. The catchall ensures that a non-essential or unrecognised command is acknowledged. Random catchall responses such as.."Sounds fun but I've got a headache" or "I did that yesterday" or "That won't achieve anything" will give more variety rather than a fixed response. I consider it vital that a variety of responses are used to make the game entertaining. You will note my examples are somewhat eccentric; you can equally use serious replies. Any author using simply "I can't do that" deserves to be shot.

The real secret to writing an adventure is structure. It's inevitable that the code is going to be lengthy and if you don't take care, a rats nest of indeciferable code will result. Keep a track of what each section and variable does. Keep a careful eye on GOSUBS. Avoid nested GOSUBs since if you get an OUT OF MEMORY ERROR it can take hours to track it down. Use REMs initially to help you keep track but *don't* make a REM the target of a GOTO or GOSUB.

In the next instalment I will discuss interpreters and moving about the scenario and will give a listing for a machine code interpreter which will provide a little zip to your adventures.

The all-singing, all-dancing software has arrived from the States. Kevin Cox went backstage.

THAT'S ENTERTAINMENT

QUESTION: WHAT HAVE THE RECORD and the software companies got in common? Frankie Goes To Hollywood is one answer — Ocean have just released a game featuring members of the band. Dave Greenfield of the Stranglers has written an adventure game for the Spectrum which is tacked onto the tail end of the cassette version of their latest album, Aural Sculpture. Even the words 'cassette' and 'disc' are inter-changeable between the two industries.

"Software is part of entertainment."

Now there are also a couple of new companies with proven track records (pardon the pun) in the music business making the move into software. They are CBS Software which grew out of CBS Records, and Ariolasoft, a part of the big German corporation, Bertelsmann which also owns Arista/Ariola Records. Both new companies insist, however, that the most important thing they have in common with the record industry is entertainment. "Software is part of entertainment," said Frank Brunger, the Sales and Marketing Manager of Ariolasoft. "It is to all intents and purposes a fun element which is what music is all about."

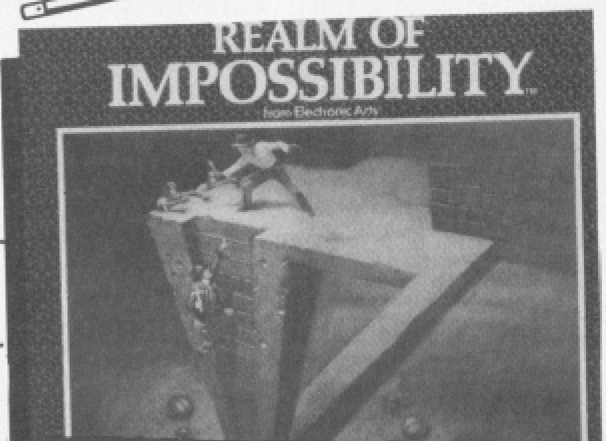
This fun element came over very strongly when I went to interview Frank, and Ashley Gray, the Managing Director. They not only share the same desk (it is rather a large one) but they also share the same sense of humour. At times you wonder if Brunger and Gray might not have made a name for themselves as a comic double act in another branch of the entertainments business. For example, I asked Frank if they personally appraised every game which is to appear under the Ariolasoft label.

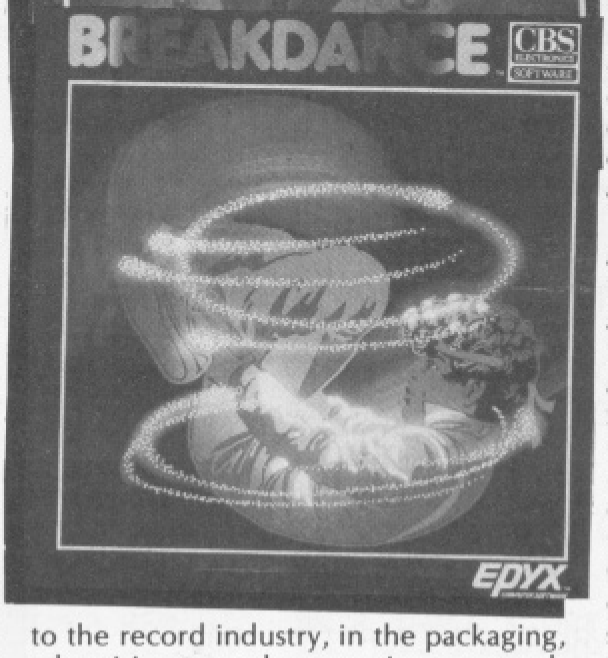
Brunger: We look at every single game on several different levels.

Gray: If we can get to them, of course.

"We have attempted to ally quite closely to the record industry."

Both men have plenty of experience of what entertains people. Until Ariolasoft was started in October last year they had spent more years that I cared to ask them working for CBS Records. They know the age group at which pop records and computer games are traditionally aimed. "We have attempted to ally quite closely





to the record industry, in the packaging, advertising etc., because it very much suits the market," said Frank. He sees games in much the same way that he used to see pop singles at CBS, though he realises that a good game may have a much longer life span than the brief chart entry of a one-hit wonder. And on the subject of charts he is adamant that the industry must adopt a single and credible chart which will become the standard — the Gallup one which appears in Your Commodore for the first time this month is the one that gets his vote.

So what about the software. Their first twelve releases are all American games for the 64 which they have re-packaged and translated into cassette versions. All of the games come from two of the most respected U.S. software companies, Broderbund and Electronic Arts and they have all been hits in America. You may already have heard of some of them: Lode Runner, Choplifter, M.U.L.E., Raid on Bungeling Bay because they have been available as imports priced at about £30-£40, roughly equivalent to their cost in the U.S. So how can Ariolasoft sell them in this country at £9.95? Ashley Gray told me, "I think the fundamental point is that if you are going to exploit the market, you have to charge the market price. The market price in the U.K. is not £30 a game. If it were there would be no room for us, because as you know Broderbund would have done it already. The product has been available on import. However, if you remove the freight, remove the customs duties and you end up with different, cheaper packaging then you are bound to end up with a cheaper price." As he explained the development costs have already been met, in the U.S.

CBS Software have a similar arrangement with the American software house, Epyx, famous for Summer Games and now the brilliant Impossible Mission. Brian Hyams of CBS admits that they arrived late in the market — the reason Summer Games was released by Quicksilva was because CBS Software was not ready at the time. However, it was only last year that the U.S. and U.K. markets found a machine in common, the Commodore 64, which made it economically feasible to import and adapt software. It is too early to say what effect importing U.S. software will have on the home market but the repercussions could be considerable. Because American games are disc based and much more expensive they are generally more highly developed than the product we are used to. I put it to Ashley Gray that we may eventually in this country become completely dependent on American software because nobody here will be able to afford the costs of longer development times needed to compete. "I disagree with you fundamentally because I think the reason U.K. software is not as good as it could be in comparison with the U.S. stuff is very simple; that is, that business as we have it in the U.K. is, in many different aspects, not yet fully developed. And I think you will find that most U.K. software houses have not yet decided to develop software which is marketable worldwide. You only have to look at the names of some of the games

and they are solely U.K. based products. They don't mean anything to anybody outside the U.K."

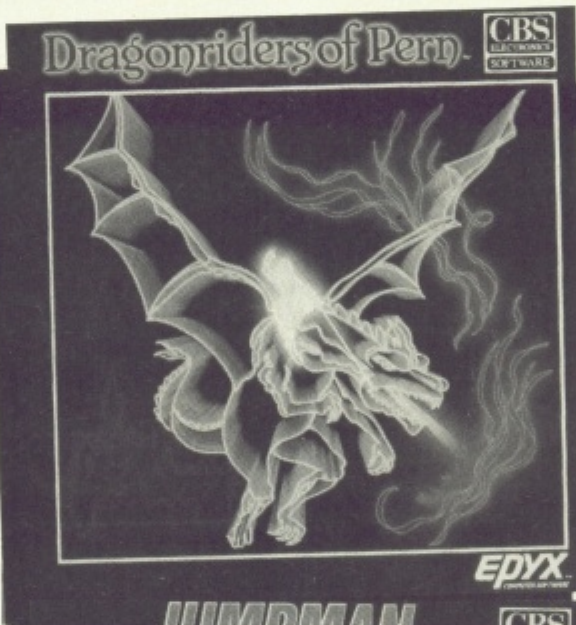
He believes that it is not necessarily greater development times which will allow U.K. software to compete but greater attention to detail which can be had for minimal extra cost. That being the case, I wanted to know if he foresaw the possibility of exporting U.K. software into the States at considerably reduced prices and perhaps cleaning up. "I think it unlikely that that will happen. I think it much more likely that, certainly as far as Ariolasoft is concerned, we will source better, more expensive software in the U.K. and sell it at a higher price in the U.S. to pay for it. And gradually the two countries will come towards an equilibrium." He added, "I fundamentally believe that if we are to create our own presence in terms of sourcing product in the U.K. we have to consciously source product which is the best and which stands the best chance of competing against the existing Ariolasoft products in their own markets. In other words, I am not interested in sourcing product which cannot stand up against Broderbund and Electronic Arts in the States."

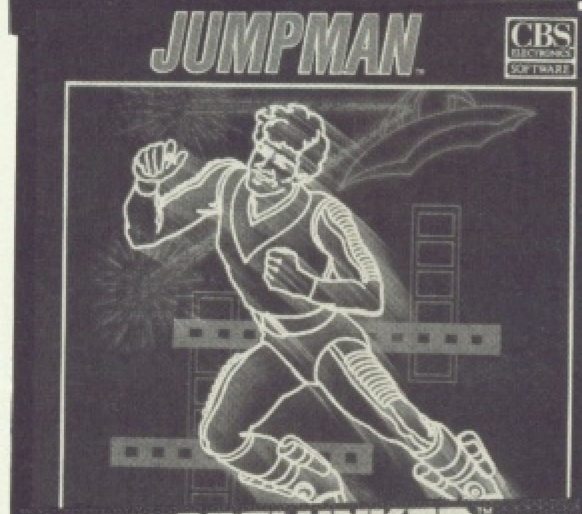
"...it should be a major piece of product which has totally international marketing possibilities."

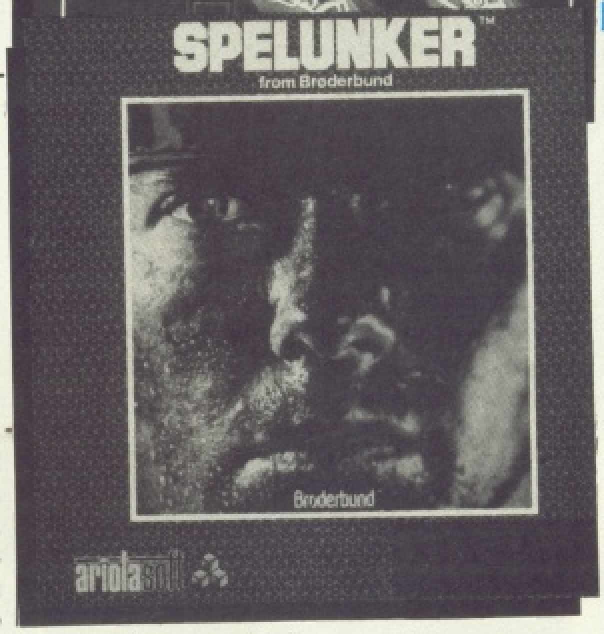
Fair enough. I suppose that there is something of the puritan in me which resists the idea of yet another American invasion sweeping aside the home-grown product. Now I know that nobody is going to buy a second-rate game just because it was written down the road and that it's a lot better on the pocket to pay £10 for a program rather than £30. However, it's still good to have the assurance that a company like Ariolasoft is committed to U.K. software.

So when can we expect the first non-American Ariolasoft game? Ashley Gray again, "Well, we have one piece of product which shall remain nameless, because we are not telling anybody about it as yet, which we hope to have available for shipping to the trade in the middle of the summer. It will be sourced through a U.K. software house. I do not know as yet where it will be written. It may very well be written in the U.K., it may be written in France, Germany or wherever." Nothing had been finalised when I spoke to him but he added, "If we go ahead with it, it should be a major piece of product which has totally international marketing possibilities. It's going to be very big." No hints, no clues? "No hints whatsoever." You can bet I'll be chasing him up about that one.

With all their other connections in the entertainments industry, I wondered how







soon they would be organising tie-ups with records as they are released and later perhaps films. First, Brian Hyams from CBS Software told me that he does not have simultaneous rights to CBS music. However, he is not considering commissioning U.K. software until the end of this year, so it may be a little early for him to say.

Ariolasoft were keen but saw the pitfalls. Tie-ups cost money and they don't always work, was Ashley Gray's line. "Ghostbusters is a good example of a company (Activision) that got it right," he said. However, he also named another couple of examples of which he was more sceptical. Nevertheless, he was aware of the potential of good music on a game, "If the sound is good, it is a big plus."

Another inheritance from the music industry was his and Frank Brunger's vehement condemnation of all forms of piracy. All software houses are united in their attack on organised piracy. However, I have spoken to some who express a certain understanding for home copying. Perhaps the companies are afraid of offending some of their customers. Ariolasoft is adamant. For them the slogan, home taping is killing music has become home taping is killing software. Frank said that he can see no distinction between going into a shop and stealing a toy or stealing software by copying it.

For this reason they are against all forms of software hire schemes and they will not allow any of their games to be down-loaded electronically, though they have yet to consider Compunet. They are also looking forward to the time when software is simultaneously launched in the U.S. and the U.K. Only then will bootlegging become unprofitable. To do this they are being, and I quote, 'fairly aggressive in clearing the back-log' of games. We should be in for a bonanza of software on the Ariolasoft label for some time to come.

"If the sound is good, it is a big plus"

I have really left the most important advantage which these two companies possess until last: experience. Both companies have a good, saleable product but they also have the ability to exploit it. You can't hype rubbish for long, but many excellent programs have disappeared for want of exposure. As Frank Brunger says, "The days of the cottage industry are passed," and he is a man who has not come from the cottage, but the glass tower world of a successful international company. Similarly, Ashley Gray says, "There has to be a new professionalism." They are both relative newcomers to what is really a very new industry, but they bring with them the disciplines of a related and more established business. Already they are looking at new ways of getting their message across (without the advantages of the record industry's ready-made medium, radio) and looking at new outlets for their product (record stores are the obvious target). Combined with this marketing experience, though, is a freshness of approach. Frank Brunger admits that he is no computer buff, but sees this as no disadvantage. "If a program works for me, it should work for the mass market."

Above all, like all good record companies, both Ariolasoft and CBS Software are not just chasing the hits but building a broadly based catalogue. The next new releases from CBS are to be educational programs, a market Ariolasoft is also intending to enter, as well as the business field under the provisional title of Prosoft.

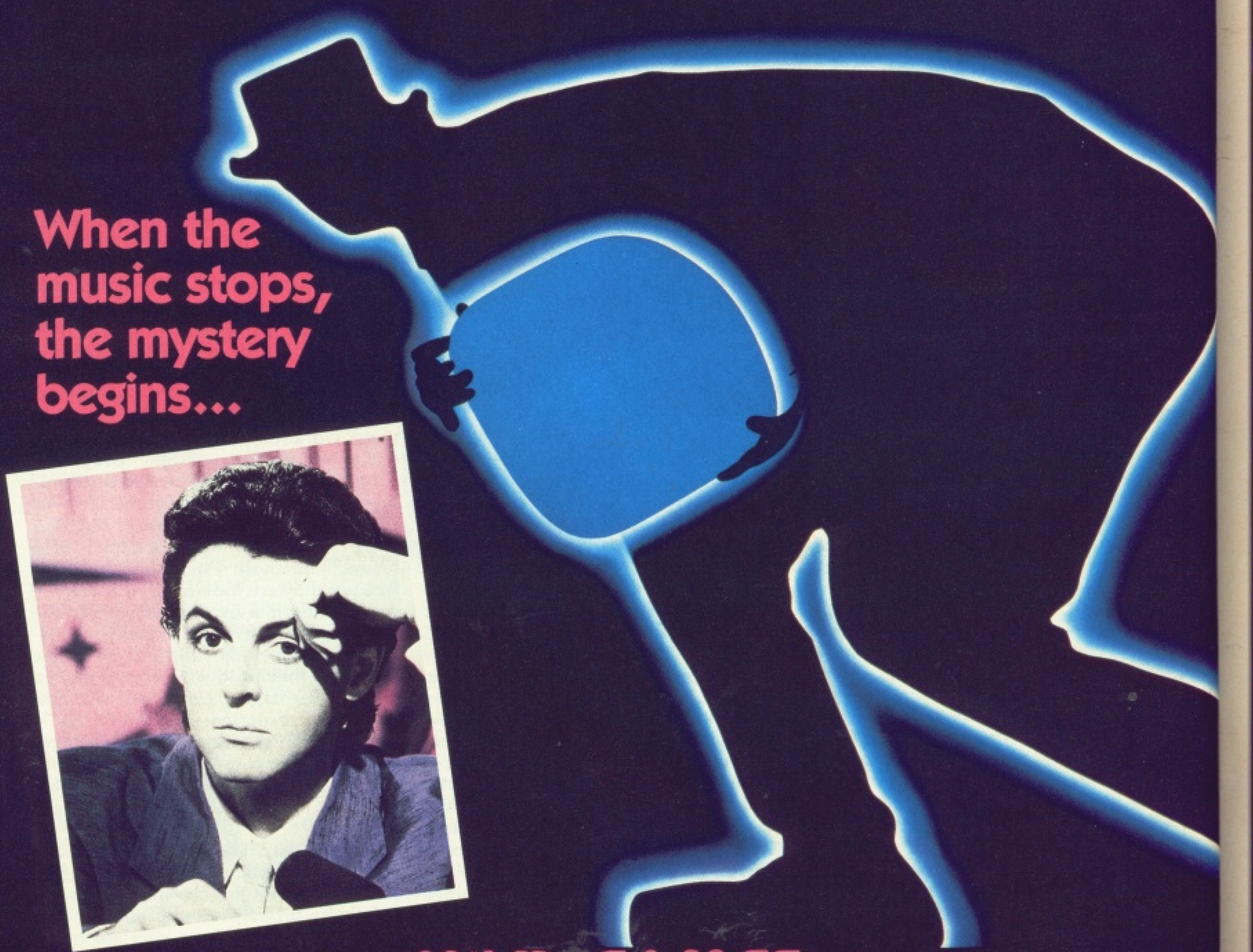
Entertainment is after all a serious business, but it can also be fun.

...7 busy characters, 10 lost chords, 12 hours, 48 Traffic Wardens, 95 London Tube Stations, 126,720 square feet of London, 7 million Londoners ... 943 action filled screens.

PAUL McCARTNEY'S *Give my regards to*

BROAD STREET

When the music stops, the mystery begins...



MIND GAMES

COMMODORE 64 – £7.99

YOUR COMMODORE

Lineage: 40p per word.

Semi display: £9.00 per single column centimetre
Ring for information on series bookings/discounts.

All advertisements in this section must be prepaid.
Advertisements are accepted subject to the terms and conditions printed on the advertisement rate card (available on request).

01-437 0699

Send your requirements to:
SALLY COLLINS
ASP LTD, 1 GOLDEN SQUARE,
LONDON W1

YOUR COMMODORE — CLASSIFIED ADVERTISEMENT — ORDER FORM

1.	2.	3.
4.	5.	6.
7.	8.	9.
10.	11.	12.
13.	14.	15.

Advertise nationally in these columns for only 40p per word (minimum charge 15 words). Simply print your message in the coupon and send with your cheque or postal order made payable to Argus Specialist Publications Ltd. to:

CLASSIFIED DEPARTMENT, YOUR COMMODORE, No. 1 Golden Square, London W1. 01-437 0699

Name ....

Address ....

....

....

Tel. No. (Day) ....

Please place my advert in **Your Commodore** for ______ issues. Please indicate number of insertions required.

THE BIG THREE

THREE ESSENTIAL PACKAGES CAN BE YOURS FOR WELL UNDER THREE FIGURES!

micro magpie

DATABASE

Micro Magpie for the Commodore 64 is probably the most advanced database management system available for any home micro. With Micro Magpie you can create a database system tailored exactly to your own information handling requirements.

With most database systems, you only get out what you put in. Micro Magpie gives you more! It can manipulate data between different files, perform calculations on numerical data, print out reports, and hardcopy data in graphical form. Only Micro Magpie can give you all these features . . .

* Fully user programmable database management system.
* Operated by easy-to-use pop-up menus using just four keys.
* 100% machine code program for speed and compactness.
* Can perform complex calculations on numerical data.
* Searches on any field, with wild card and part-field matching.
* Works with one or two 1541 disk drives.
* Horizontal or vertical bar graph or scatter graph output.
* Integral Centronics interface for parallel printer option.
* Professionally written instruction manual.
* Help readily available from Help Disk.
* Free application templates - Mailing List and Stock Control.

Micro Magpie would be excellent value if you bought it for the applications alone! You can have a highly sophisticated Mailing List system up and running within minutes, and you'll still be using only a fraction of Micro Magpie's potential! Once you start tailoring your own custom applications, you will begin to discover why Commodore User magazine called Micro Magpie "the software bargain of the year"!

MICRO MAGPIE - ON DISK ONLY

£39.95 inc VAT

Micro Swift

SPREADSHEET

Micro Swift - the affordable professional spreadsheet system for the Commodore 64. Micro Swift will prove indispensable in business or around the home to construct models of income and expenditure; for carrying out instant 'what-if' calculations to see how a change in one or more figures affects all the other figures; or for specialised applications where complex number crunching is required.

Micro Swift is unique in that it is operated by pop-up menus, a system pioneered by our Magpie database program. No longer do you have to memorise a whole hierarchy of commands and control keys - the menus appear when you want them, and disappear when you don't.

Micro Swift is written totally in machine code (beware other spreadsheets in Basic!) for speed and compactness, giving you more room to construct more complex models.

Micro Swift gives you these amazing features . . .

* A1 to Z254 cell matrix.
* Variable cell widths.
* Split screen facility.
* User definable numerical precision and display formatting.
* Integral Centronics Interface for parallel printer option.
* Automatic execution of user-programmed sequences.
* Graphical display option.
* Operated by easy-to-use pop-up menus.
* Free ready-programmed applications included.
* File storage on disk or cassette.
* Full instruction manual included.

Don't be fooled by the price! Micro Swift gives you facilities equal, if not superior, to spreadsheets costing many times more. Micro Swift is available on disk or cassette. Micro Swift - Spreadsheet power to the people!

MICRO SWIFT - ON DISK OR CASSETTE

£19.95 inc VAT

MICRO WORDCRAFT

WORD PROCESSOR

Micro Wordcraft is our new disk-based professional word processor. It is a direct descendant of the highly respected Wordcraft program, which is in use world-wide on the larger Commodore machines, Sirius, and IBM PC, etc. Written in 100% machine code, Micro Wordcraft spells affordable office quality word processing for the home or business user.

It is often said that "you get what you pay for"; but with the Audiogenic Professional Series you get what you pay for and more!

Micro Wordcraft gives you all these advanced facilities . . .

* Full text control - document width up to 99
* columns, tabs, decimal tabs, justification and centering.
* Full text manipulation - on-screen editing, block move, block delete, string search and
* replace, underlining and emboldening.
* Scrolling screen display, uncluttered by control characters.
* Name and address files can be created and merged into standard letters.
* Easy merging of standard paragraphs. Compatible with Commodore, parallel and RS232 printers.
* Integral Centronics interface for parallel printer option. Instantly accessible Help screens.
* Comprehensive instruction manual included.

The name and address merging capabilities of Micro Wordcraft make it an ideal tool for small businesses, clubs, societies or hobby groups, where there are regular mailings of standard letters. For home use, Micro Wordcraft contains all the facilities you could ever need, at a price you can easily afford!

MICRO WORDCRAFT - ON DISK ONLY

£24.95 inc VAT

The Audiogenic Professional Series represents a price breakthrough for business-orientated software products. With Micro Magpie, Micro Swift, and Micro Wordcraft, the power and convenience of the computerised office can be a reality for all Commodore 64 owners!

Each of the three products represents the state of the art in its particular field. Great care has been taken over the documentation to make sure that you, the user, can quickly make use of the facilities available. Also, a full backup service is provided by the Audiogenic Technical Department, who are only a phone call away if you have any queries about the products.

So, whether it's for your home, business, club or society, make sure you go for the Audiogenic Professional Series!

Audiogenic LTD

PROFESSIONAL SERIES

AUDIOGENIC LTD. P.O. BOX 88, READING, BERKS. ENGLAND. Tel: (0734) 664646

Choosing the right computer is a good start — but can you find the right software?



At SUPERSOFT we're very conscious of the fact that people who spend several hundred pounds on computer equipment are looking to do rather more than play Space Invaders.

Financial planning is a rather grand name for something you've been doing all your life – making ends meet! Perhaps if Mr Micawber had used **BUSICALC** he would have been able to balance the books a little better.

For home, club or small business use BUSICALC 1 should pay for itself in no time at all; for larger companies we recommend BUSICALC 3, one of the few really valuable programs that you can learn to use in a day.

Although your Commodore 64 is a powerful musical instrument you need to be a pretty good programmer to understand how it all works. Unless, of course, you buy **MUSIC MASTER!**

To use MUSIC MASTER requires no prior musical knowledge, though in the hands of an experienced musician it will prove an invaluable tool. You don't need to know the first thing about programming either! MUSIC MASTER is the musical equivalent of a word processor, remembering the notes you play and allowing you to replay and edit them as you wish.

INTERDICTOR PILOT is a space flight simulator. Nowadays simulators are widely used to train pilots and astronauts because – to be frank – it's a lot cheaper (and safer) than the real thing!

Imagine, if you will, life in the 22nd century: space travel is commonplace, and on the outskirts of the galaxy the first war between civilizations is being fought. A shortage of trained pilots has prompted the Federation to develop a computer simulation that allows raw recruits to gain experience without paying for their mistakes with their lives. With the aid of your Commodore 64 you too can learn to pilot the Interdictor Mk 3 craft. But be warned – this is no game!

SUPERSOFT

SUPERSOFT, Winchester House, Canning Road, Wealdstone, Harrow, Middlesex HA3 7SJ
Telephone: 01-861 1166

Other SUPERSOFT products include the **MIKRO ASSEMBLER** cartridge, the only assembler that's ideal for beginners yet powerful enough for the professional (most of our competitors use it!). The **VICTREE** cartridge adds dozens of commands to Basic including toolkit aids and disk commands; or on disk there's **MASTER 64,** a really comprehensive package for the keen programmer.

Of course, we do also publish games programs, and with classics like **STIX, QUINX** and **KAMI-KAZE** in our range we are one of the market leaders. But we most enjoy coming up with the sort of programs that are going to be in use for months and years, not hours and days – the sort of programs that make you glad that you bought a computer – and glad that you bought SUPERSOFT!

You won't find SUPERSOFT products on the shelves of your local supermarket. But most specialist shops stock titles from our extensive range (and are prepared to obtain other programs to order). However you can also buy direct by sending a cheque (pre-paid orders are post free!), by calling at our offices, or over the telephone using your ACCESS card.